Statistical Arbitrage in South African Equity Markets

Khuthadzo Masindi

Student no : MSNKHU003

Supervisors : Sugnet Lubbe and Kevin Kotze

Dissertation presented for the degree of Master of Philosophy in Mathematical Finance

Faculty of Commerce

University of Cape Town

17 February 2014

Thesis

Khuthadzo Masindi

October 6, 2014

Plagiarism declaration

1. I know that plagiarism is wrong. Plagiarism is to use anothers work and pretend that it is my own.
2. I have used the APA referencing guide for citation and referencing. Each contribution to, and quotation in this thesis from the work(s) of other people has been contributed, and has been cited and referenced.
3. This thesis is my own work.
4. I have not allowed, and will not allow, anyone to copy my work.

Signature :

Date : 17 February 2014

Publication

I hereby grant the University free license to publish this dissertation in whole or in part, and in any format the University deems fit

Acknowledgements

I would like to express my special appreciation and thanks to my supervisors Professor Sugnet Lubbe and Mr Kevin Kotze, your assistance has been priceless. A special thanks to my family. Words cannot express how grateful I am to my father, my mother and my sister for your support and prayers. I would also like to thank all of my friends who supported me, and encouraged me to strive towards my goal.

Abstract

The dissertation implements a model driven statistical arbitrage strategy that uses the principal components from Principal Component Analysis as factors in a multi-factor stock model, to isolate the idiosyncratic component of returns, which is then modelled as an Ornstein Uhlenbeck process. The idiosyncratic process (referred to as the residual process) is estimated in discrete-time by an auto-regressive process with one lag (or AR(1) process). Trading signals are generated based on the level of the residual process.

This strategy is then evaluated over historical data for the South African equity market from 2001 to 2013 through backtesting. In addition the strategy is evaluated over data generated from Monte Carlo simulations as well as bootstrapped historical data. The results show that the strategy was able to significantly out-perform cash for most of the periods under consideration. The performance of the strategy over data that was generated from Monte Carlo simulations demonstrated that the strategy is not suitable for markets that are asymptotically efficient.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Investigations into the behaviour of various security prices have evolved from early Babylonian times to a point where we currently consider the application of sophisticated techniques on various markets [1].Over this extended period of time, various studies have been conducted by a large number of interested parties, including academics, financial participants, governments and industry experts; which has resulted in the generation of diverse body of research.

Financial markets have also evolved over time and they currently make use of a wide variety of financial instruments, including complicated derivatives that are traded daily. Techniques for the application of risk management strategies have also evolved; assisted by the proliferation of financial modelling techniques. In addition, a large number of practitioners in the computational finance field have sought to develop models that seek to profit from securities that may be mis-priced at specific periods of time.

It is important to note that the idea that particular securities may be mispriced at particular points in time is not altogether inconsistent with the efficient market hypothesis (EMH), which maintains that market prices fully reflect the current information set that is available to the public.[1] For instance as suggested by [5], whilst market efficiency could be the norm, it is quite likely that there will be temporary departures from absolute efficiency,

[1]For contrasting views on the applicability of [2]'s influential survey of the EMH, see "A Random Walk Down Wall Street" by [3] and "A Non-Random Walk Down Wall Street" by [4].

particularly during market bubbles and crashes.[2]

Furthermore as noted by [6], whilst there is no a priori reason why regularities in asset price dynamics should not exist, it is likely that any easily identifiable effects will soon be eliminated or arbitraged away in the very process of being exploited. However, it may be possible to identify opportunities in the form of previously undiscovered behaviour, by looking at the information set that is provided by markets from a new perspective that takes advantage of advances in computational finance.

To identify such behaviour, this dissertation seeks to implement a computational model that utilises historical security prices to generate trading strategies that exploit the nature of the evolution of securities prices. A major motivation for applying a new technique to financial data is that the data generating processes of financial and other economic time series are at best imperfectly understood. Therefore, the use of new computational techniques may offer individuals the possibility of identifying opportunities that may not have been considered.

In terms of the practical applications, this dissertation extends the work of [7], which uses Principal Component Analysis to model a portfolio of securities. The residuals from this model are then used to model the non-systematic components of the stocks in the portfolio; where these idiosyncratic returns are modelled as mean-reverting Ornstein-Uhlenbeck processes. This modelling procedure allows for the derivation of a trading rule that is evaluated on the basis of the risk adjusted returns that would have been generated over time. The robustness of these results, which are generate for securities on the JSE over the period from December 2001 to December 2013, are then investigated with the aid of Monte-carlo and bootstrapping techniques.

The following chapters include a brief review of pure arbitrage and statistical arbitrage. Thereafter, a pairs trading strategy is introduced as a simple example of statistical arbitrage. The application of principal component analysis is then presented as a computational tool that is used to further develop the pairs trading idea into a generalised factor trading strategy. Subsequently a simple mean reversion model for generating trading signals is proposed. This leads to a discussion on trading rules and trading

[2]Of course, as financial participants trade on such potential instances they may trade away certain inefficiencies, thus making markets more efficient and making the EMH true asymptotically.

signals, before the trading strategy is evaluated based on historical data. The final chapter contains the conclusion.

Notation and Definitions

n : the total number of stocks
m : the total number of days in a data sample
p : the number of principal components
$\mathbf{R}_1, \ldots, \mathbf{R}_n$ be the stock return vectors

$$\bar{R}_i = \sum_{j=1}^{m} \mathrm{R}_{ji}$$

$$\bar{s_i} = \sqrt{\frac{1}{m-1} \sum_{j=1}^{m} \left(R_{ji} - \bar{R}_i\right)^2}$$

$$\mathrm{A}_{ji} = \frac{R_{ji} - \bar{R}_i}{\bar{s_i}}$$

$\mathbf{v}_k$:the k^{th} eigenvector
$\mathbf{Q}_k$: the k^{th} eigenportfolio
s_i : the s-score corresponding to stock i

Chapter 2

Arbitrage

Arbitrage is a term with different meanings, the definition accepted by practitioners is that it is trading that seeks to make low-risk profits by exploiting discrepancies in the price of the same asset or in the relative prices of similar assets [8]. A classic example exists in the markets of homogeneous commodities such as the respective world Gold markets. When the price of Gold in Johannesburg exceeds the price in London by more than the costs of transport (and other logistical considerations), then a trader can earn arbitrage profit by selling Gold for delivery in Johannesburg whilst simultaneously buying Gold in London to be delivered to Johannesburg. In this case the trader's profit is considered to be "risk free".

According to [9], there are three types of arbitrage opportunities in the real world.

- Pure arbitrage - an opportunity that allows an investor to earn more than the risk free rate without the risk of loss. Pure arbitrage opportunities rarely exist in the market, when they occur they do not last for long, due to competition amongst traders.
- Near arbitrage - near arbitrage is based on the law of one price. The law of one price states that if two portfolios have the same payoffs at some future date, then the value of the two portfolios must be equivalent. Practitioners seek to profit by trading in assets that replicate each of the respective assets' cash flows (or payoffs) despite trading with significant price differences. The practitioners are exposed to risk in this endeavour as there is no guarantee that the prices will converge.
- Speculative arbitrage - in speculative arbitrage investors take advan-

tage of what they view as mispriced assets based on some prior relationship that exists between the assets, one could develop a speculative arbitrage opportunity based on the correlation structure between two assets for instance. Statistical arbitrage could be considered as speculative arbitrage.

2.1 Pure Arbitrage

An arbitrage in this case is a portfolio value process $X(t)$ satisfying $X(0) = 0$ and also satisfying for some time $T > 0$

$$\mathcal{P}\{X(T) \geq 0\} = 1 \tag{2.1}$$

$$\mathcal{P}\{X(T) > 0\} > 0 \tag{2.2}$$

In an arbitrage trade the agent starts with no capital and at a later time (T) she has a positive probability of earning returns in excess of the risk free rate without any probability of losing money. Such an opportunity exists, if and only if there is a way to start with positive capital $X(0)$ and earn a return greater than that of a money market account [10]. Consequently arbitrage exists if there is a way to start with a positive $X(0)$, and at a later time T have a portfolio value that satisfies

$$\mathcal{P}\{X(T) \geq e^{rT}X(0)\} = 1 \tag{2.3}$$

$$\mathcal{P}\{X(T) > e^{rT}X(0)\} > 0 \tag{2.4}$$

where e^{rT} is the compound factor

This definition of arbitrage is prevalent in academia topic, the formulation is foundational in arbitrage pricing theory.[1].

2.2 Near Arbitrage

Near arbitrage opportunities can be characterised as those opportunities that

[1]Most foundational Stochastic Calculus for Finance including [11] and [12] cover this topic in depth

arise from appropriate "lock-in" transactions. That is buying the underpriced asset whilst simultaneously selling the appropriate overpriced asset which is usually some kind of a replicating portfolio of the first asset. Given an asset X_t and a replicating portfolio $\mathrm{R}(X_t)$, those opportunities can be represented as

$$|\text{payoff}(X_t - \mathrm{R}(X_t))| > \text{TransactionCost} \tag{2.5}$$

Index arbitrage is a good example of "risk-less" arbitrage. It occurs between the equities constituting a particular market index and the associated index forward contract. If F_t is the index forward price, S_i is the price of a constituent stock, w_i is the weighting of a constituent stock, r and q denote the risk free interest rate and the dividend yield respectively then

$$|F_t - \sum_i w_i S_t^i \exp^{(r-q_i)(T-t)}| > \text{TransactionCost} \tag{2.6}$$

is an example of a "near arbitrage" opportunity[6].

Another good example is the so called put-call parity relationship. If S is the stock price and P is the price of a put option with strike K and maturity T, and C is the price of a call option also with a strike K and maturity T, with r denoting the continuously compounded risk free rate then the put-call parity relation for European options is given by

$$|C + Ke^{rT} - P - S| > \text{TransactionCost} \tag{2.7}$$

Opportunities of this type are obviously highly sought after, as a result they are very scarce, and when they do exist in the market they only do so momentarily, due to the actions of arbitrageurs. Consequently only those who are geared to trade very rapidly and with very low transaction costs earn a reasonable return from "risk-less" arbitrage [2].

It is important to note that the term "risk-less" arbitrage does not always imply the absence of any financial risk. For example basis risk may occur when there is a mismatch between a position and its hedge. This may sometimes lead to losses. Interestingly enough, arbitrage opportunities exist from basis risk, and it is the belief that the basis between an asset and its replicating portfolio reverts to zero that drives arbitrage transactions. Liquidity risk, which is the risk that a position cannot be closed within a desired time frame

[2]See [13] for further discussion of put call parity and the associated arbitrage opportunities

is also a hazard for arbitrageurs. Near arbitrage is really just pure arbitrage in practice as the pure arbitrage definition is strictly applicable only under very idealised conditions.

2.3 Statistical arbitrage

Statistical arbitrage is used as a type of speculative arbitrage. It can be defined as a portfolio value process $X(t)$ satisfying $X(0) = 0$ and for some time $T > 0$

$$E[X(T)] > 0, \qquad \mathcal{P}\{X(T) > 0\} > 0 \tag{2.8}$$

Note that in statistical arbitrage the agent expects to make money at time T but there is a possibility of also losing money at time T.

Studies suggest that simple statistical arbitrage opportunities do not exist or only exist under idealised market conditions. For example it has been shown that if a statistical arbitrage opportunity is defined as an opportunity where (i) the expected payoff is positive and (ii) the conditional expected payoff in each final state is non-negative and the pricing kernel in the market is path independent, then no such statistical arbitrage opportunities can exist[14].

In practice, pairs trading is considered a classic example of statistical arbitrage. In the next chapter a statistical arbitrage framework is developed from pairs trading.

Chapter 3

Pairs trading and portfolio construction

Pairs trading or statistical arbitrage trading is loosely defined as trading one financial instrument (or basket of financial instruments) against a second financial instrument (or basket of financial instruments) where one may go long in the one instrument(s) and short in the other(s).

Pairs trading strategies were popularized in 1985 when a small group of quantitative researchers at the investment bank Morgan Stanley created a program to buy and sell stocks in pair combinations. The Morgan Stanley Black Box as it was called, quickly earned a reputation and a lot of money[15]. This prompted other investment houses to employ similar strategies during that period of time.

A further appealing feature of this strategy is that it can be extremely simple to apply. After identifying a pair of stocks that exhibit a similar historical price trajectory, one would need to determine when the relative price spread between the two stocks exceeds a set threshold. The agent would buy the under-performing stock whilst simultaneously selling the stock that has over-performed. The strategy will earn a profit if the spread were to narrow or reverse. These pairs trading strategies generally have low exposure to systematic market shocks as the agent would have simultaneous long and short positions in stocks that are fundamentally related as such the strategy is also usually market neutral with returns that are uncorrelated with the returns of the market.

The chart below shows the performance of Continental Airlines share price

(CAL) and American Airlines share price (AMR) between 2002 and 2004[1]. Note that the spread between the two shares narrows and widens over time. In hindsight it is obvious that investors would have made a profit by shorting the over performing security when the gap widens and simultaneously buying the under performing security.

Figure 3.1: Daily performance indices (2002 - 2004)

Today most pairs trading strategies are strongly quantitative and they usually rely on rules based investment decisions. Such investment decisions or trading signals are generated by a computer program using statistical and/or mathematical techniques. The chart below illustrates how trading rules operate in the presence of a mean reverting spread between two pairs.

[1]This example was adopted from [15]

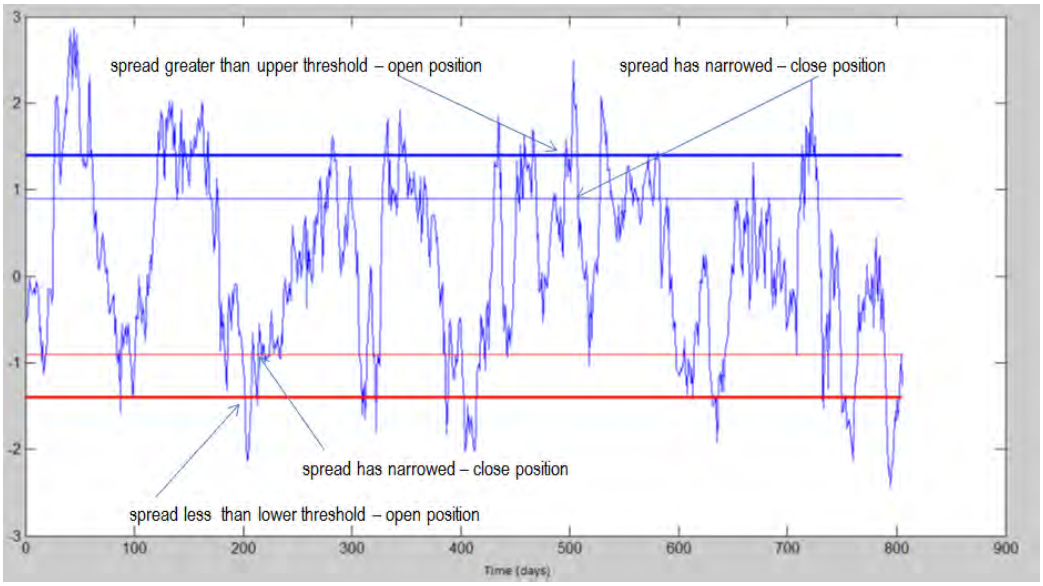

Figure 3.2: Mean reverting spread between a pair

A position is opened when the spread exceeds a set threshold by selling the relatively overpriced stock and buying the relatively under priced stock. The positions are closed when the spread narrows significantly, but this is not necessarily when the spread becomes zero. Positions tend to be closed earlier to limit risk.

To formalise the strategy let P and Q be the prices of two fundamentally related stocks. The log returns from an arbitrary strating point t_0 to time t is given by $\ln\left(\frac{P(t)}{P(t_0)}\right)$ and $\ln\left(\frac{Q(t)}{Q(t_0)}\right)$ respectively. In the absence of random noise, the two stocks would change price over time in an equivalent manner which can be modelled by $\ln\left(\frac{P(t)}{P(t_0)}\right) = \alpha(t - t_0) + \beta \ln\left(\frac{Q(t)}{Q(t_0)}\right)$.

Deviations from this relationship between the two stocks can be due to noise and is modelled by an additional term such that $\ln\left(\frac{P(t)}{P(t_0)}\right) = \alpha(t - t_0) + \beta \ln\left(\frac{Q(t)}{Q(t_0)}\right) + X(t)$. This relationship between the two stock in differential form is:

$$\frac{dP(t)}{P(t)} = \alpha dt + \beta \frac{dQ(t)}{Q(t)} + dX(t) \tag{3.1}$$

In this case $X(t)$ is a mean-reverting process or the *residual* process, β is the correlation coefficient and α is the drift coefficient. Equation 3.1 models a

linear relationship between returns P and Q. Typically the intercept term αdt is small relative to the residual process $X(t)$, which models the random noise in the spread between P and Q and can be ignored. Therefore, after controlling for β a portfolio which is long P and short Q can be modelled as the mean reverting process $X(t)$. Note that equation 3.1 can be estimated using ordinary linear square regression.

This trading strategy can be expanded to more than two stocks, let P_1, P_2 and P_3 be the prices of three stocks, where the relationship between the stocks can be modelled as,

$$\frac{dP_1(t)}{P_1(t)} = \alpha dt + \beta_2 \frac{dP_2(t)}{P_2(t)} + \beta_3 \frac{dP_3(t)}{P_3(t)} + dX(t) \tag{3.2}$$

The addition of a third share into this model introduces new challenges which are highlighted below:

- Whilst equation 3.1 can be estimated using ordinary least square regression (OLS), however, due to the possibility of a strong correlation between P_2 and P_3 in the equation, there may be a problem with heterogeneity if equation 3.2 is estimated from ordinary linear square regression.

- In contrast with pairs trading where one would select fundamentally related stocks, there is no obvious way of selecting an additional share from our universe of available shares whilst still maintaining a mean reverting spread.

In addition to the above, a general problem with pairs trading is that there is a limited supply of fundamentally related shares in the market and pairs trading is a very popular strategy, so any easily identifiable opportunities are quickly traded away as soon as they appear.
Given the above challenges, one could wish to make use of exchange traded funds (ETFs) which represent an investment portfolio of stocks or bonds or commodities. ETFs are traded on the exchange. There are for instance ETFs that track a broad stock index such as FTSE/JSE All Share index. Other stock market ETFs may track stocks in a particular sector, or class of securities.

One may be able to solve the problem of a limited supply of pairs in a market through the use of ETFs in place of the shares P_2 and P_3. ETFs also simplify the process of identifying pairs. For example an ETF tracking the

broad stock index F_1 could be used in place of P_2 and F_2 the ETF that is tracking P_3's sector. In this case Equation 3.2 then becomes,

$$\frac{dP_1(t)}{P_1(t)} = \alpha dt + \beta_2 \frac{dF_1(t)}{F_1(t)} + \beta_3 \frac{dF_2(t)}{F_2(t)} + dX(t)$$

This is re-written below as,

$$\frac{dP_1(t)}{P_1(t)} = \alpha dt + \beta_1 \frac{dF_1(t)}{F_1(t)} + \beta_2 \frac{dF_2(t)}{F_2(t)} + dX(t) \tag{3.3}$$

As noted above one could make use of two ETFs (being an ETF tracking the market and an ETF tracking an appropriate sector), however, by doing so the problem of heterogeneity would make the use of ordinary least square untenable. This is because an ETF tracking the entire market is likely to be correlated with an ETF tracking a particular sector. Consequently the model would have to be restricted to just one ETF, such an ETF would need to track the subject share (given by P_1 in equation 3.2) closely. Finding such an ETF might be an issue due to the dearth of sector ETFs listed on the Johannesburg Stock Exchange (JSE).[2]

As an alternative to the use of ETFs, Principal Component Analysis (PCA) can be used to remedy the above shortcomings. PCA is a procedure that uses the variance covariance matrix of a data set containing possibly correlated variables to create new variables known as principal components (or factors). The principal components consist of uncorrelated linear combinations of the original set.

Since PCA is constructed such that the first principal component account for as much of the variation as possible, the second, as much of the remaining variation and so on, only a few principal components account for the majority of the variation in the original data. In this way PCA allows one to simplify the data structure into a few components.

The components from PCA could be used to express the common factors in a portfolio of assets in a multi-factor model. Therefore the ETF factors in equation 3.2 could be replaced by the principal components $(F_1(t), F_2(t), \ldots, F_p(t))$. Furthermore the model's explanatory power could be adjusted to a desired

[2]The ETFs available on the JSE only track broad sectors such as Financials, Industrials and Resources. There are no ETFs tracking basic materials or telecommunication stocks for example.

level by adding or withdrawing components. To express the model, let $(F_1(t), F_2(t), \ldots, F_p(t))$ be the principal components where p is chosen to yield a desired explanatory power. Then equation 3.3 becomes,

$$\frac{dP_1(t)}{P_1(t)} = \alpha dt + \beta_1 \frac{dF_1(t)}{F_1(t)} + \beta_2 \frac{dF_2(t)}{F_2(t)} + \cdots + \beta_p \frac{dF_p(t)}{F_p}(t) + dX(t) \quad (3.4)$$

This technique has many desirable properties and in the the following section an overview of the mathematics of PCA is proposed to illustrate how PCA can be used to construct a portfolio of stocks.

3.1 Principal component analysis

Let $\mathbf{S}_1, \ldots, \mathbf{S}_n$ be historical security price vectors for a market with n securities and $\mathbf{R}_1, \ldots, \mathbf{R}_n$ be the corresponding return vectors. Let us also assume that this data is available daily for m historical days with $m >> n$ so that $\mathbf{R}_{m\times n} = \big(\mathbf{R}_1, \ldots, \mathbf{R}_n\big)$ with each $\mathbf{R}_i$ an $m \times 1$ vector.

A transformation of the return data $\mathbf{R}_{m\times n}$ can be performed so that each column has a mean of zero and a standard deviation of one, thereby forming a new return matrix which will be denoted as $\mathbf{A}$. Note that each column of $\mathbf{A}$ is a series whose entries are each given by,

$$\mathrm{A}_{ji} = \frac{R_{ji} - \bar{R}_i}{\bar{s}_i} \quad (3.5)$$

where $\bar{R}_i$ is the sample mean of $\mathbf{R}_i : m \times 1$ given by $\bar{R}_i = \sum_{j=1}^{m} \mathrm{R}_{ji}$ and $\bar{s}_i$ is the sample standard deviation of $\mathbf{R}_i$ given by $\bar{s}_i = \sqrt{\frac{1}{m-1} \sum_{j=1}^{m} \left(R_{ji} - \bar{R}_i\right)^2}$.

The Principal Components are created from the Singular Value Decomposition of a matrix. This is shown below for the matrix $\mathbf{A}_{m\times n}$

$$\mathbf{A}_{m\times n} = \mathbf{U}_{m\times n}\mathbf{D}_{n\times n}\mathbf{V}^{\mathbf{T}}_{n\times n} \tag{3.6}$$

where $\mathbf{D} = diag(d_1, ..., d_n)$. The d_i's are the singular values of $\mathbf{A}$. They are ordered so that $d_1 \geq d_2 \geq \cdots \geq d_n \geq 0$ whilst $\mathbf{U}$ and $\mathbf{V}$ are chosen to have orthonormal columns. Now we have

$$\begin{aligned}\mathbf{A}^T\mathbf{A} &= (\mathbf{U}\mathbf{D}\mathbf{V}^T)^T(\mathbf{U}\mathbf{D}\mathbf{V}^T)\\ &= \mathbf{V}\mathbf{D}\mathbf{U}^T\mathbf{U}\mathbf{D}\mathbf{V}^T\\ &= \mathbf{V}\mathbf{D}^2\mathbf{V}^T\end{aligned}$$

so that

$$(\mathbf{A}^T\mathbf{A})\mathbf{V} = \mathbf{V}\mathbf{D}^2 \tag{3.7}$$

the last line follows because $\mathbf{U}$ has orthonormal columns and $\mathbf{D}$ is diagonal whilst we have $\mathbf{V}^T = \mathbf{V}^{-1}$. It follows from the above that the columns of $\mathbf{V}$ must be the eigenvectors of $\mathbf{C} = \mathbf{A}^T\mathbf{A} = (n-1)\mathbf{var(A)}$ which is also the covariance matrix bar a factor of $m-1$. The singular values and eigenvalues of $\mathbf{C}$ are both d^2. The singular values of $\mathbf{A}$ are the square roots of the corresponding eigenvalues of $\mathbf{C}$.

It can be shown that the linear combination $\mathbf{P}_1 = \mathbf{Ab}$ with maximum variance is obtained when $\mathbf{b} = \mathbf{v}_1$, the first column of $\mathbf{V}$ corresponding with the largest eigenvalue of $\mathbf{C}$. Similarly the maximum variance, uncorrelated with $\mathbf{P}_1$, is obtained for $\mathbf{Ac}$ when $\mathbf{c} = \mathbf{v}_2$, the second column of $\mathbf{V}$ corresponding with the second largest eigenvalue of $\mathbf{C}$ [16].

Therefore

$$\mathbf{P}_k : m\times 1 = \mathbf{A}\mathbf{v}_k = \begin{bmatrix}\mathbf{A}_1 \dots \mathbf{A}_n\end{bmatrix}\begin{bmatrix}v_k^{(1)}\\ \vdots \\ v_k^{(n)}\end{bmatrix} = \mathbf{A}_1 v_k^{(1)} + \cdots + \mathbf{A}_n v_k^{(n)}$$

Since $\mathrm{A}_{jk} = \frac{R_{jk}-\bar{R}_k}{s_k}$ where $\bar{R}_k$ is the sample mean $\mathbf{R}_j : m\times 1$ and $\bar{s_k}$ is

the sample standard deviation of $\mathbf{R}_j$, it follows that

$$P_{jk} = v_k^{(1)} \times \frac{R_{j1} - \bar{R}_1}{\bar{s_1}} + v_k^{(2)} \times \frac{R_{j2} - \bar{R}_2}{\bar{s_2}} + \cdots + v_k^{(n)} \times \frac{R_{jn} - \bar{R}_n}{\bar{s_n}}$$

$$= v_k^{(1)} \times \frac{R_{j1}}{\bar{s_1}} + v_k^{(2)} \times \frac{R_{j2}}{\bar{s_2}} + \cdots + v_k^{(n)} \times \frac{R_{jn}}{s_n} - \left(v_k^{(1)} \times \frac{\bar{R}_1}{\bar{s_1}} + v_k^{(2)} \times \frac{\bar{R}_2}{\bar{s_2}} + \cdots + v_k^{(n)} \times \frac{\bar{R}_n}{\bar{s_n}} \right)$$

Since the second term above is independent of j, the different principal components all contain the same constant $-\left(v_k^{(1)} \times \frac{\bar{R}_1}{\bar{s_1}} + v_k^{(2)} \times \frac{\bar{R}_2}{\bar{s_2}} + \cdots + v_k^{(n)} \times \frac{\bar{R}_n}{\bar{s_n}} \right)$. In the remainder of the dissertation the new variables as illustrated below will be used

$$\mathbf{Q}_{(k)} : m \times 1 = \begin{bmatrix} v_k^{(1)} \times \frac{R_{11}}{\bar{s_1}} + v_k^{(2)} \times \frac{R_{12}}{\bar{s_2}} + \cdots + v_k^{(n)} \times \frac{R_{1n}}{\bar{s_n}} \\ \vdots \\ v_k^{(1)} \times \frac{R_{m1}}{\bar{s_1}} + v_k^{(2)} \times \frac{R_{m2}}{\bar{s_2}} + \cdots + v_k^{(n)} \times \frac{R_{mn}}{\bar{s_n}} \end{bmatrix}$$

This is re-written below as

$$\mathbf{Q}_{(k)} : m \times 1 = \begin{bmatrix} v_k^{(1)} \times \frac{R_1(1)}{\bar{s_1}} + v_k^{(2)} \times \frac{R_2(1)}{\bar{s_2}} + \cdots + v_k^{(n)} \times \frac{R_n(1)}{\bar{s_n}} \\ \vdots \\ v_k^{(1)} \times \frac{R_1(m)}{\bar{s_1}} + v_k^{(2)} \times \frac{R_2(m)}{\bar{s_2}} + \cdots + v_k^{(n)} \times \frac{R_n(m)}{\bar{s_n}} \end{bmatrix} \tag{3.8}$$

An eigenportfolio is a portfolio whereby the weightings of the securities are determined by the eigenvectors. Therefore from equation 3.8, the return time series for the k^{th} eigenportfolio is given by

$$\mathbf{Q}_k : m \times 1 = \sum_{i=1}^{n} v_k^{(i)} \frac{\mathbf{R}_i}{\bar{s_i}} \tag{3.9}$$

Let $\lambda_k = d_k^2$ be the k^{th} eigenvalue corresponding to the k^{th} eigenvector of $\mathbf{C}$, [16] further shows that the proportion of the variance in the data that can be explained by the k^{th} principal component is given by

$$\frac{\lambda_k}{\sum_{i=1}^{n} \lambda_i} \tag{3.10}$$

3.2 Residual Process

The purpose of this section is to present a model that can be used to decompose stock returns into systematic factors and an idiosyncratic factor.

As shown above, the value of the k^{th} eigenportfolio at a point in time is given by,

$$F_k(t) = \sum_{i=1}^{n} \frac{v_k^{(i)}}{\bar{s}_i} S_i(t) \tag{3.11}$$

This follows from 3.9 which shows that the returns for investing in a portfolio where the weighting of each stock is determined by the eigenvector. Thus to get the rand value of the portfolio, the returns can be substituted directly by the prices for each stock. Equation 3.11 is used to compute the historical price data for the k^{th} eigenportfolio from the historical data of stock prices. Thus allowing the formulations that follow to be in continuous time.

Let p be the number of principal components required to adequately explain the variance in stock market returns, also let n be the number of securities available in the market, the change in the stock price should be proportional to the change in the p eiegportfolios; i.e

$$\frac{dS_i(t)}{S_i(t)} \approx \beta_i \frac{dF(t)}{F(t)} = \sum_{k=1}^{p} \beta_{ik} \frac{dF_k(t)}{F_k(t)}$$

Here β_{ik} is the sensitivity of stock i to the eigenportfolio k. This can be expanded by including the drift αdt and the idiosyncratic term $dX(t)$. This leads to the model given by

$$\frac{dS_i(t)}{S_i(t)} = \alpha_i dt + \sum_{k=1}^{p} \beta_{ik} \frac{dF_k(t)}{F_k(t)} + dX_i(t) \tag{3.12}$$

The model given by equation 3.12 is a multi-factor model. [7] found that the first eigenportfolio tracks the market capitalisation weighted portfolio closely when they applied this model to the constituent stocks of the S&P Index.

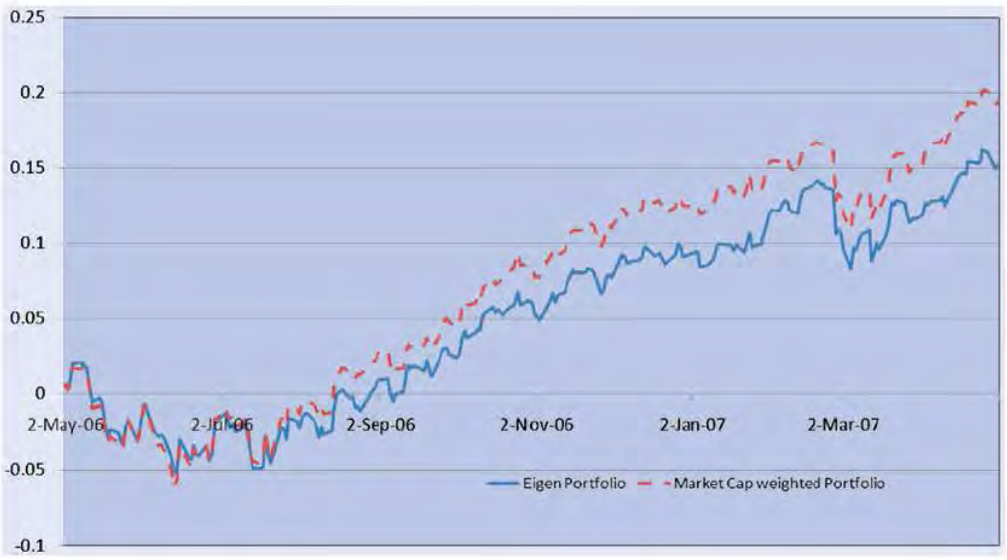

Figure 3.3: Comparative evolution of the first eigenportfolio and the market weighted portfolio

They also found that when the coefficients of the k^{th} eigenvector are ranked in decreasing order so that

$$v_k^{(1)} \geq v_k^{(2)} \cdots \geq v_k^{(n)}$$

where $v_k^{(i)}$ is the i^{th} coefficient in the k^{th} eigenvector. Note that the coefficients have been reordered such that $v_k^{(1)}$ is the largest and $v_k^{(n)}$ is the smallest coefficient in the eigenvector. Recall that each coefficient in an eigenvector is associated with a stock or company, and that the weighting of each stock in an eigenvector can be derived from the coefficient. After re-ordering [7] found that the neighbours of a particular stock (those companies with similar sized coefficients) tended to be in the same industry group. This property which they called coherence held true for high ranking eigenvectors.

To summarise this section, equation 3.12 uses the eigenportfolios to explain returns. The first eigenportfolio normally tracks the performance of a market weighted portfolio. The other high ranking eigenportfolios are overweight shares in a particular industry so they are similar to industry factors. In addition these factors are uncorrelated to each other, which is an outcome of PCA.

The terms $\alpha dt + dX_i(t)$ in the multi-factor model explain price fluctuations,

which are not driven by the high ranking eigenportfolios. In particular $dX_i(t)$ is referred to as the idiosyncratic component of returns, which is of great importance in this dissertation.

Essentially after identifying the idiosyncratic component of returns,(the *residual process*) in the model we may be able to derive trading signals. Such a residual process may be modelled as a mean reverting process where trading signals can be generated using the variation of the process from its mean.

It is important to note that the model in equation 3.12 relates the value of the i-th stock S_i with the eigenportfolios. For the remainder of the dissertation, the discussion that follows will be in the context of 3.12, further note that the discussions in this dissertation is limited to generating trading signals for a given stock i, versus the eigenportfolios (as opposed to trading portfolios of stocks against each other).

3.3 Ornstein-Uhlenbeck process

The following section provides an overview of the Ornstein-Uhlenbeck (OU) process which is the stochastic process that was used to model the residual process $dX_i(t)$. The OU process was first proposed in a paper by Uhlenbeck and Ornstein in 1930. It was suggested as an alternative to the Brownian motion in instances where a mean reverting tendency is needed. The model has been used in a wide variety of applications including interest rate modelling in Mathematical Finance and is discussed in several texts that cover Stochastic Calculus techniques. Footnote (See Shreve for a comprehensive discussion on the use of the processes). The mean reverting OU equation is given by

$$dX_i(t) = \kappa_i(m_i - X_i(t))dt + \sigma_i dW_i(t), \quad \kappa_i > 0 \tag{3.13}$$

where m_i is the mean reversion level and κ is the speed of reversion The conditional expectation of the process is given by,

$$\begin{aligned}
\mathrm{E}\big[dX_i(t)\big|X_i(s), s \leq t\big] &= \mathrm{E}\big[\kappa_i(m_i - X_i(t))dt + \sigma_i dW_i(t)\big|X_i(s)\big] \\
&= \kappa_i(m_i - \mathrm{E}[X_i(t))|X_i(s)]dt \\
&= \kappa_i(m_i - X_i(s) - \mathrm{E}[X_i(t) - X_i(s))|X_i(s)]dt
\end{aligned}$$

$$= \kappa_i(m_i - X_i(s))dt$$

This follows because the OU process has independent increments which means that the process is expected to increase whenever its value $X_i(t)$ is less than the mean reverting level m_i, and vice versa.

The analytical solution to equation 3.13 is given by

$$X_i(t_0 + \Delta t) = e^{-\kappa_i \Delta t} X_i(t_0) + (1 - e^{-\kappa_i \Delta t}) m_i + \sigma_i \int_{t_0}^{t_0+\Delta t} e^{-\kappa(t_0+\Delta t-s)} dW_i(s) \tag{3.14}$$

where the mean is given by,

$$\mathrm{E}[X_i(t_0 + \Delta t)] = e^{-\kappa_i \Delta t} X_i(t_0 + (1 - e^{-\kappa_i \Delta t}) m_i \tag{3.15}$$

The variance for a process may then be expressed as,

$$\mathrm{Var}[X_i(t_0 + \Delta t)] = \frac{\sigma_i^2}{2}(1 - e^{-2(t_0+\Delta t)}) \tag{3.16}$$

If we let $\Delta t \to \infty$ then we may have the equilibrium probability that is normally distributed with a mean given by,

$$\mathrm{E}[X_i(t_0 + \Delta t)] = m_i \tag{3.17}$$

and a variance

$$\mathrm{Var}[X_i(t_0 + \Delta t)] = \frac{\sigma_i^2}{2\kappa_i} \tag{3.18}$$

Now because the conditional expectation of the process is given by

$$\mathrm{E}\big[dX_i(t) \big| X_i(s), s \leq t\big] = \kappa_i(m_i - X_i(s))dt \tag{3.19}$$

X_i is expected to increase whenever its value is less than the mean reverting level m_i, and vice versa. This means that trading signals can be created based on this expectation, i.e buy X_i when it is less than m_i and vice versa. To avoid excessive trading, a buffer can be included such that the trading signal becomes; buy when $X_i - m_i > B$, where B is chosen appropriately.

The average size of the spread, X_i, will vary for each trade depending on the valuation. Some trades will have a mean reverting spread that is very

narrow, whilst others will have a spread that is generally wide. Therefore, one choice for the value of the buffer, B, may be satisfactory for one trade whilst being inappropriate for another. Determining a suitable value for B_i can be problematic and is usually solved by defining a new dimensionless variable s_i known as the s-score[3].

$$s_i = \frac{X_i - m_i}{\sigma_i}$$

The absolute average size of the spread is then taken into account in the trade signal, through the inclusion of the variance of the residuals in the denominator.

3.4 Estimating the OU model

The Ornstein-Uhlenbeck process can be estimated in discrete time as an auto-regressive process with one lag (which would be equivalent to a discrete time AR(1) process). Before the AR(1) model can be estimated, the idiosyncratic component $dX_i(t)$ has to be isolated from the systematic factors. This is done by regressing the returns of share i against a set of systematic (explanatory) factors which in this dissertation are the eigenportfolios.

In a market with n securities the OLS model for stock i is given by

$$\mathbf{R}_i = \beta_0 + \sum_{k=1}^{p} \beta_{ik} \mathbf{Q}_k + \boldsymbol{\epsilon} \tag{3.20}$$

Where $\mathbf{R}_i$ is a column vector of the returns of stock i, $\mathbf{Q}_k$ is a column vector of the returns of the k^{th} eigenportfolio and p is the number of eigenportfolios. The residual process is then defined from the fitted regression, as

$$X_{ij} = \sum_{h=1}^{j} \hat{e}_h, \quad j = 1, 2, \ldots, m \tag{3.21}$$

where m is the length of the regression model estimation window. The residual process can be modelled as the AR(1) process given by

$$X_{i,j+1} = a + bX_{ij} + \zeta_{i,j+1} \tag{3.22}$$

[3]The definition of the s-score has been adopted from [7]

which is the discrete version of equation 3.15. All that remains now is to estimate the parameters, where the residual process is given by

$$X_i(t_0+\Delta t) = \underbrace{(1-e^{-\kappa_i \Delta t})m_i}_{a} + \underbrace{e^{-\kappa_i \Delta t}}_{b} X_i(t_0) + \underbrace{\sigma_i \int_{t_0}^{t_0+\delta t} e^{-\kappa_i (t_0+\Delta t-s)} dW_i(s)}_{\zeta}$$

The parameters a, b, ζ are estimated from the AR process. These estimates are then used to solve the system of equations for the three unknowns m, κ and σ. The solution for m, κ and σ is shown in Appendix A.

Chapter 4

Formulating the Investment Strategy

This section provides an overview of the methodology followed when formulating the investment strategy. In particular, this section focuses on the trade generation process, including a discussion on trading costs and the effect of leverage.

4.1 Signal generation

4.1.1 Introduction

Investors buy shares that they believe are undervalued and they sell shares that they believe are overvalued. The proposed trading strategy must generate a buy signal when a stock is undervalued relative to the eigenportfolios and vice versa.

If the drift term is ignored, equation 3.12 can then be re-written as

$$dX_i(t) = \frac{dS_i(t)}{S_i(t)} - \sum_{k=1}^{N} \beta_{ik} \frac{dF_k(t)}{F_k(t)} \tag{4.1}$$

$dX_i(t)$ can also be seen as the profit and loss from a trade that is long in stock i and short in the eigenportfolios. In essence, the residuals of the regression model equation in equation 3.20 represent the mispricing at each time [17].

It is important to note that the eigenportfolios are effectively tradable as they represent a portfolios of shares.

If the model determines that $X_i(t)$ which is the residual process will be increasing, then the trading signal should be to purchase share i and sell the eigenportfolios. Recall, that it is expected that X_i will be increasing whenever $X_i < m_i$ and vice versa, since $X_i(t)$ is a mean reverting process. This procedure is illustrated in figure 4.1 which shows the evolution of the s-score and the level of the s-score for opening and closing trades.

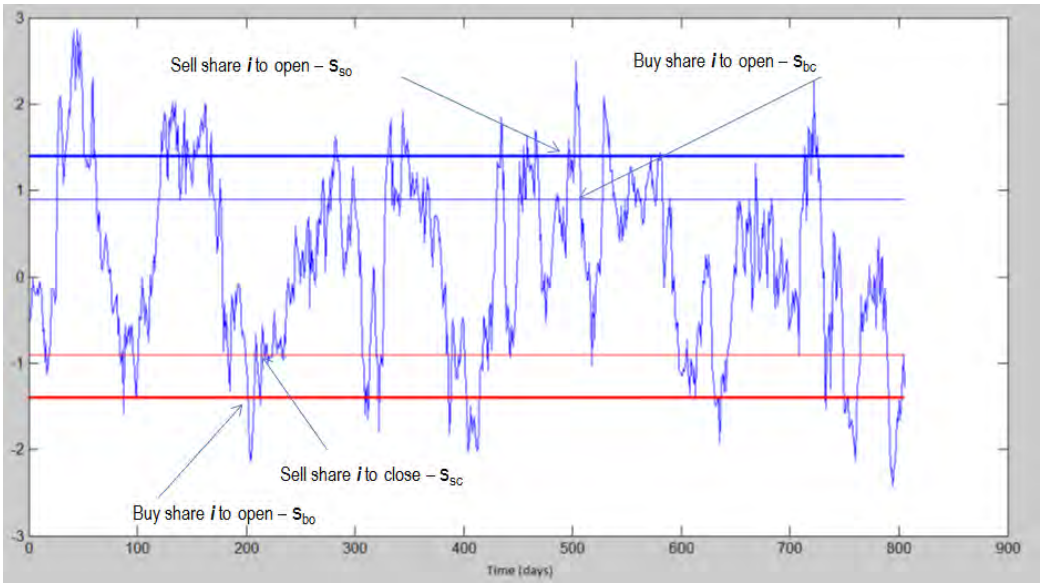

Figure 4.1: Mean reverting spread between a pair

The basic signal for trading can be defined as

$$\text{buy to open if } s_i < -\bar{s}_{bo}$$

$$\text{sell to open if } s_i > +\bar{s}_{so}$$

$$\text{close long position when } s_i > -\bar{s}_{sc}$$

$$\text{close short position when } s_i < +\bar{s}_{bc}$$

where the trading signals s_{bo}, s_{so},s_{sc} and s_{bc} are determined through backtesting simulations. This is discussed later in the dissertation.

4.1.2 Profit and loss and trading signals

The discussion in the previous sub-section was concerned with modelling the evolution of the regression residual process (and by extension the s-score), before a trade has been opened. In this sub-section further details of the evolution of the residual process after a trade has been opened is discussed to develop more intuition around this investment strategy. In particular we seek to pay attention to the modelling of the process after a trade has been opened.

Note that the residual process captures the non-systematic difference between the rate of return on stock i, which is given by $\frac{dS_i(t)}{S_i(t)}$ and the rate of return on a weighted portfolio of eigenportfolios, which is in turn given by $\sum \beta_{ik} \frac{dF_k(t)}{F_k(t)}$. Hence the evolution of the residual process before a trade has been open is determined from the regression errors and the evolution after a trade has been open is determined by the trade profit and loss.

Given the importance of the profit and loss, it would be optimal to incorporate this information in the modelling of the residual process after a trade has been opened. The profit and loss data could be used in modelling the residual process as a feedback mechanism.

When every trade is open we have $\hat{X}_i = 0$. This is because $\hat{X}_i$ is defined by equation 3.21 as the sum of the residuals. A property of ordinary linear square regression is that it forces the sum of the residuals to zero when an intercept term is included in estimating the regression.

Recall that equation 3.21 gives the original regression process. Now let d denote the number of days for which a trade is open, and also let ω_q be the profit and loss earned on the q-th trading day. Then after a trade is open, the continued residual process (i.e the residual process incorporating profit and loss) is given by,

$$X_i = \overbrace{\sum_{j=1}^{m} \hat{e}_j}^{(i)} + \overbrace{\sum_{q=1}^{d} \omega_q}^{(ii)} \tag{4.2}$$

where part (i) is the original regression residual process given by equation 3.21, note that part (i) of equation 4.2 does not change after a trade has been opened, also note that its value is zero when a trade is open, part (ii) is the cumulative profit and loss process. The cumulative profit and loss process

is the difference between the returns of share i after a trade is open, which is given by $\frac{dS_i(t)}{S_i(t)}$ and the return of the eigenportfolios after a trade is open, which is given by $\sum \beta_{ik} \frac{dF_k(t)}{F_k(t)}$.

It is important to note that the cumulative profit and loss process is defined similarly to the residual process. The only differences being that the residual process is estimated from the regression errors and the cumulative profit and loss process is the realised profit and loss. We therefore have an option of using the cumulative profit and loss process as a feedback mechanism. This is discussed in more detail in the sub-section that follows.

Recall, that at the inception of a trade, where share i is undervalued relative to the eigenportfolios, we have at time t_0 an s-score that is less than $-\bar{s}_{bo}$, such that,

$$s_i = \frac{X_i(t_0) - m_i}{\sigma_{\text{eq},i}} < -\bar{s}_{bo} \text{ but } \hat{X}_i = 0 \text{ at } t_0 \text{ so}$$

$$s_i = \frac{-m_i}{\sigma_{\text{eq},i}} < -\bar{s}_{bo}$$

The trade will be closed at time T and when s-score is greater than $-\bar{s}_{sc}$ so

$$s_i = \frac{X_i(T) - m_i}{\sigma_{\text{eq},i}} > -\bar{s}_{sc}$$

After closing the trade, the change in the s-score s_i is given by

$$\begin{aligned} \Delta s_i &= \frac{X_i(T) - m_i}{\sigma_{\text{eq},i}} - \frac{-m_i}{\sigma_{\text{eq},i}} \\ &= \frac{X_i(T)}{\sigma_{\text{eq},i}} \\ &= \bar{s}_{bo} - \bar{s}_{sc} \end{aligned}$$

So that

$$X_i(T) = (\bar{s}_{bo} - \bar{s}_{sc})\sigma_{\text{eq},i} \tag{4.3}$$

According to equation 4.2, the residual process X_i gives the profit and loss of a trade. The process X_i in equation 4.2 is the same as $X_i(T)$ in equation 4.3, this shows that the profit and loss that can be generated from a trade is proportional to the difference between the cut-off points. In general the choice of cut-off levels s_{open} and s_{close} determines the maximum possible profit and

loss for the model, and more importantly, we have established that there is direct relationship between the trading signals and profit and loss.

Part (ii) of equation 4.2 shows that the residual process changes after a trade is opened, due to profit and loss. This means that the parameters, σ_{eq} and m_i, are not constant if part (ii) is incorporated into the model (as the parameters are estimated from the residual process). Therefore as noted before, when implementing the strategy there is an option that utilises constant parameters that are estimated from the regression or alternatively one could use using dynamic parameters by incorporating part (ii) of equation 4.2 which uses the cumulative profit and loss process as a feedback effect. In the next subsection we discuss the different possible implementations of the above strategy.

The following flow diagram provides a summary of the trade generation process.

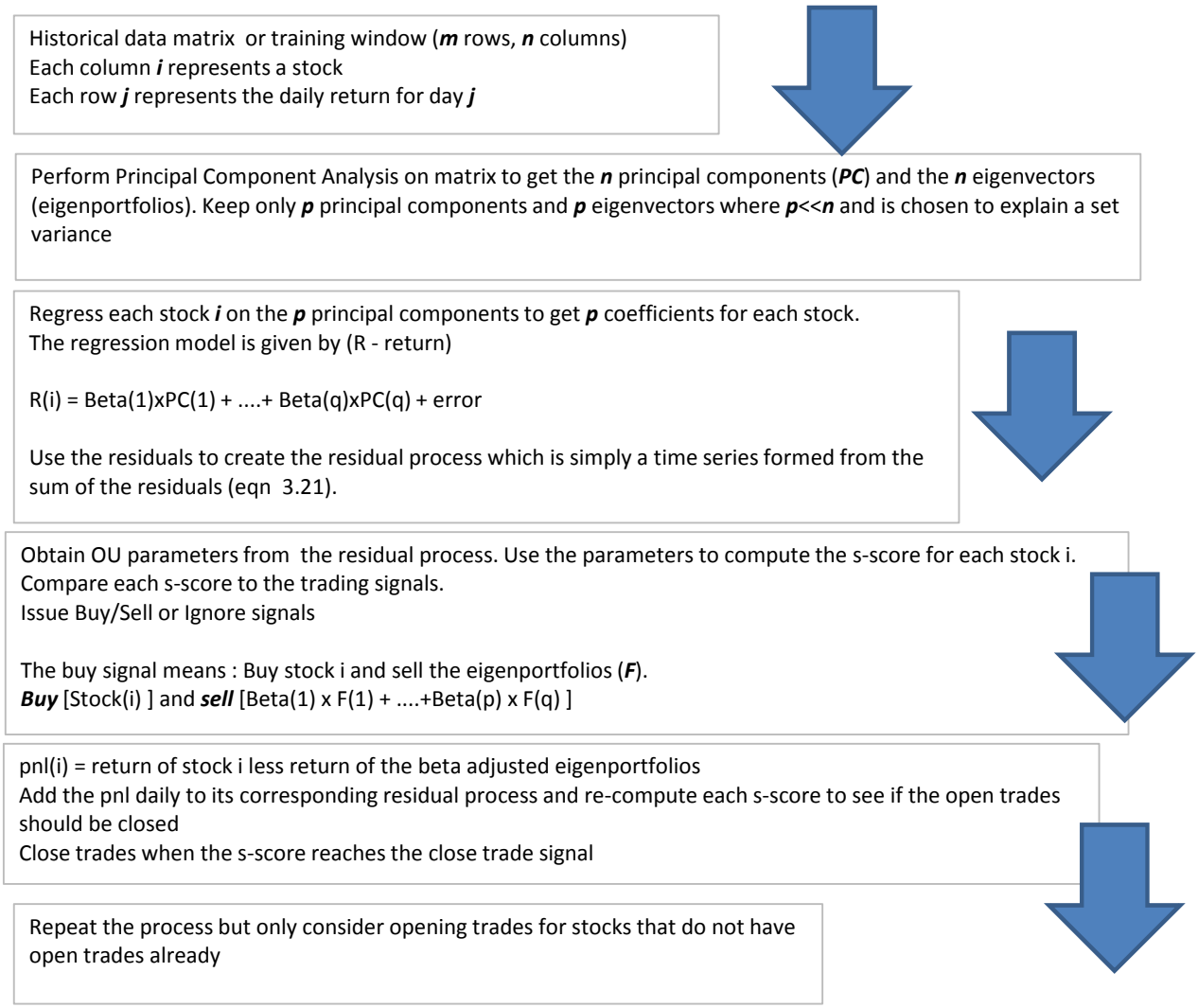

Figure 4.2 : Summary of the trade generation process

4.2 Implementing the residual process model

The purpose of this section is to discuss the framework that was followed when estimating the parameters on which trades are based.

The majority of active investors seek to predict security returns over various time horizons. They would then trade to profit from those predictions. However securities markets are highly dynamic and as such most investors would need to update their expected returns on a regular basis. Consequently the optimal portfolio evolves as expectations are revised. [1]

In a world without transaction costs, investors would trade regularly to ensure that their holdings match that of their optimal portfolio. However, in reality, excessive trading may detract from the performance of an investment strategy. This means that investors have to find a trade-off between holding the optimal portfolio and incurring transaction costs. This is of greater importance when implementing strategies that by nature require frequent rebalancing, for example high frequency trading, statistical arbitrage and index tracking.

To this effect, the persistence of a trading signal is important. Persistent signals are signals that are slow to change, for example a signal to buy share i and sell the eigenportfolios should only change when the value of share i relative to the eigenportfolios has increased sufficiently to earn the investor their required return. When trading signals are persistent then investors are able to minimise the amount of trading required to re-balance their portfolios.

Persistent signals tend to have a slow mean reversion.. However if two investments have the same expected return (alpha) - fast mean reversion is preferable to slow mean reversion. The aim here is to avoid getting into lots of trades that have small alphas (potential expected returns) but very slow mean reversion. This is because in that case the proportion transaction costs as a percentage of the profit will be very high. The transaction costs will likely erode any gains from such a strategy. In addition a slow reversion

[1] Institutional investors often refer to a "model portfolio" as their optimal portfolio. It is the responsibility of the implementation team to ensure that the model portfolio is suitably replicated for clients.

would result in high relative holding costs.

The following trading principles were formulated from the closed form solution of an optimal trading strategy by [18]. (a) aim in front of the target and (b) partially trade towards the current aim, this is analogous to other fields of application such as missile guidance (towards a moving target), hunting and sports.

These principles are espoused in this dissertation when considering the estimation of the parameters in the residual process. Three options were considered in this regard:

(a) Keep the parameters constant by not incorporating the cumulative profit and loss process (or the newly realised residual process).

(b) Fully incorporate the entire residual process by keeping all of the original regression residual process and also the cumulative profit and loss process i.e updating the residual process without replacement.

(c) Incorporate the cumulative profit and loss process whilst simultaneously truncating the earlier parts of the original regression residual process i.e updating the residual process with replacement.

Overall the extent to which the cumulative profit and loss process is incorporated will determine the extent to which the parameters will be dynamic.

In what follows the three options as discussed above have been formulated.

4.2.1 Constant OU parameters derived from the original regression residual process

In this case the parameters obtained from the original regression residual process are static. This is because the parameters are not updated once after a trade is opened as part (ii) of equation 4.2 is discarded. Consequently a trade is only closed once the required profit, given by equation 4.3, has be earned.

Its worth noting that the trading signals from this implementation are persistent in the sense that trades are only closed when the required profit has

been earned, which keeps the portfolio turnover to a minimum.

A key disadvantage of this approach is that the profit and loss process is not used as a feedback mechanism. Once a trade is open it can only be closed when the required profit has been earned. As such, this implementation is not flexible, additionally it is not possible to trades partially towards the aim when there is no feedback mechanism.

A stop loss would usually have to be implemented to limit losses from failed trades. To minimise the reliance on a stop loss for ending failed trades, it is desirable to have an implementation that has the ability to respond to changes in market conditions. This is not possible if the model has constant parameters over the life of the trade.

4.2.2 Dynamic OU parameters derived from a residual process that fully incorporates the continued process and the regression residual process

In this case the residual process incorporates part (ii) of equation 4.2 where

$$X_i = \sum_{j=1}^{m} \epsilon_j + \sum_{q=1}^{d} \omega_q$$

This implies that the OU parameters are no longer constant after a trade has been opened. As a result, the profit and loss earned from a trade using this approach is not perfectly explained by equation 4.3. An advantage of this approach is that the profit and loss is used as feedback, which means that the strategy is able to respond to changes in market conditions. However, because this implementation fully incorporates the original regression residual process it may be slow to respond. This implementation is somewhat flexible. In addition, the implementation allows for trades to be closed before the maximum possible profit and loss (given by equation 4.3) has been earned (it is able to trade partially towards the aim).

4.2.3 Dynamic OU parameters with updating and replacement

In this case the residual process from the regression is updated with the profit

and loss such that,

$$X_i = \sum_{j=d}^{m} \epsilon_j + \sum_{q=1}^{d} \omega_q$$

This approach is very similar to the approach discussed in sub-section 4.2.2 above with one key difference, the original regression residual process is truncated sequentially (i.e if a trade has been open for 10 days, the first 10 days of the original residual regression residual process will be discarded when the parameters are estimated). This means that the longer a trade stays open, the more the strategy would rely on the profit and loss data when estimating the parameters that determine whether a trade should stay open or be closed, as the residual process is very quickly dominated by profit and loss.

The trading signals from this approach are the least persistent. However, an advantage of this approach is that it has a strong feedback mechanism which is useful when there are unexpected events in the market. This implicit risk management quality makes it more preferable to the first two implementations. It is an improvement on the approach discussed above in as far as flexibility and the ability to trade partially towards the aim.

It is important to note that a stop loss would still be employed, but only as a last resort. This implementation is able end trades that have gone wrong before a stop loss is hit, since it updates the parameters to reflect the most recent information, this information is then used to re-evaluate trades that have been open.

4.3 Parameter estimation and sample selection

The length of the historical data that is used to specify the parameters is referred to as the training window, the training data is the data set pertaining to the training window. Trading strategies that exploit long term relationships between securities would usually have longer training windows; so that the behaviour of the long term relationships could be established by the model. Consequently, the choice of the training window is in part determined by the nature of the trading signals (i.e long-term or short-term)

To capture current trading conditions, investors would usually select a train-

ing data set that includes the most recent data when specifying parameters, this means that the training window would be selected on a rolling basis. The OU model requires the following parameters for each trade (each trade is long(short) share i and short(long) the eigenportfolios)

- the drift α_i
- the speed of mean reversion κ_i
- volatility σ_i
- the mean of the residual process m_i

The following variables also have to be estimated.

- trading signals, signals to open and close trades i.e. s_{bo}, s_{so},s_{bc}, s_{sc}
- optimal training window size for performing principal component analysis
- optimal training window size for obtaining OU model parameters
- appropriate stop loss limit
- minimum required speed of mean reversion before a trade can be opened

4.4 Capital allocation

The purpose of this section is to describe the capital allocation process as well as the manner in which performance is measured. Let E_t represent the equity in the portfolio, and Q_{it} represent the capital invested in a stock i where trading costs are ϵ basis points, then capital growth is given by

$$E_{t+\Delta t} = E_t + E_t r \Delta t + \sum_{i=1}^{n} Q_{it} R_{it} - \left(\sum_{i=1}^{n} Q_{it} \right) r \Delta t + \sum_{i=1}^{n} Q_{it} D_{it} / S_{it} - \sum_{i=1}^{n} |Q_{i(t+\Delta t)} - Q_{it}| \epsilon \tag{4.4}$$

- The first term in the equation, E_t, represents the equity invested in the portfolio at time t.

- Q_{it} represents the investment in stock i
- r represents the money market rate so that $\sum Q_{it} r \Delta t$ is the borrowing charge and $E_t r \Delta t$ is the interest earned from placing the portfolio equity on deposit.Therefore the effective cost of borrowing is the difference between the interest on the equity invested in the money market account given by $E_t r \Delta t$ and the interest charge on the amount allocated to trading $\sum Q_{it} \Delta t$.
- $\sum |Q_{i(t+\Delta t)} - Q_{it}| \epsilon$ represents the trading costs incurred by the strategy including the cost of rebalancing. Trading costs are proportional to the capital invested in each share.
- $Q_{it} = E_t \Lambda$ where Λ is the leverage ratio. A leverage ratio of 100% means that the sum of the absolute value of the long and short positions is 100%.

It is important to note that the value of the initial investment at each period grows by the return earned from trading stocks (using the strategy outlined above). Net borrowing costs as well as trading costs incurred offset the returns earned from trading stocks. Equation 4.4 also shows that leverage enhances performance when the return from the trades is higher than the total costs of borrowing and trading. The leverage ratio Λ is chosen so that the portfolio has a desired level of leverage.

When considering the cost of trading, it is worth noting that it has decreased substantially over recent times. The decrease has largely been driven by the increasing the use of technology as investors are now able to place trades through the broker's computer system allowing them to bypass the brokerage desks. For large capitalisation stocks, like the stocks in the JSE Top 40 index, round trip trading costs of lower than 10 basis points can be achieved by some institutional investors.

Note that each trade consists of a long (short) position in a share, along with a short (long) position in the eigenportfolios and that the eigenportfolios are made up of shares, as a result, we are able to net off the long and short positions for each trade to form a portfolio of shares for each trade (the portfolio can also be seen as a vector where each element corresponds to the weighting of share in the portfolio). Additionally the different trades can be netted off daily to form a unified portfolio (this portfolio then represents the

orders that would be placed in the market).

In order to ensure that there are no trades that have a disproportionate influence on the unified portfolio, each trade is unitised by using the norm of the vector corresponding to the trade. If this is not done, there is a risk that trades which involve subject shares (share i) that have large factor sensitivities would dominate the return of the unified portfolio which would make the performance of the strategy more volatile.

4.5 Backtesting

Backtesting is a process of applying a trading strategy or analytical method to historical data to see how accurately the strategy would have predicted actual results. The backtesting experiments should be conducted in a manner that best reflects how the investment strategy would be effected in practice thus the portfolio holdings for time t of the backtesting simulation should be constructed by using only the information that was available before time t, that is trading decisions must be made at time $t-1$.

It is desirable that each of the parameters and variables required for the investment strategy is estimated on a rolling basis when performing backtesting. However, in order to manage the computational time that would be required for backtesting, it was decided that only some of the parameters would be estimated on a rolling basis. To this end only the OU model parameters were estimated on a rolling basis whilst the rest of the variables such as trading cut-offs, optimal size of training windows etc. were estimated once.

An initial data sample was selected so that the variables that are to be estimated on a once-off basis could be estimated and used across the backtesting sample. This initial time series data sample predates the time series data sample that was used for the backtesting simulations, so as to preserve the merit of the backtesting results. The results of the backtesting simulations are discussed in chapter 5.

4.6 Control experiments

It may be difficult to draw too many conlusions from backtesting as the

market price generating process is complex, .This is because historical data incorporates both known and unknown dynamics. Therefore in addition to backetsting, Monte Carlo simulations and bootstrapping techniques were used when evaluating the model.

4.6.1 Monte Carlo simulations

A Monte Carlo simulation is a useful way for generating return data. The simulations utilise geometric Brownian motions to generate returns that are correlated. Data that is generated from Monte Carlo simulations is idealised, which means that the investment strategy can be tested under a controlled environment. Returns generated from Monte Carlo simulations have independent increments. The independent increments condition is not always satisfied by real world stock price dynamics. Another advantage of simulating data is that it is possible to control for correlations and variances; thereby making it possible to study the model under various market conditions. The equation below shows the diffusion model that was used to generate correlated stock price returns.

$$d\mathbf{S} = \vec{\mu}\mathbf{S}dt + \vec{\sigma}\mathbf{S}d\mathbf{W} \tag{4.5}$$

where $\mathbf{S}$ is the stock price vector, $\vec{\mu}$ is the drift vector, $\vec{\sigma}$ is the covariance matrix and $\mathbf{W}$ is a vector of independent Brownian motions [13].

It has been shown by [13] that $E(S_T) = S_0 e^{\mu T}$. This means that the expected return from a long position in a share is the drift μ. In this idealised world, a long-short investment strategy should yield a return of zero (assuming that average number of long positions is equal that of short positions and that there are no transaction costs). Therefore it is expected that investment strategy (also long-short strategy) outlined in this dissertation would perform poorly in a Monte Carlo world as trading costs would lead to negative returns on average.

4.6.2 Bootstrapping

Another useful technique for generating a time series is with the use of Bootstrapping techniques that can be used to sample historical time series data. The moving block bootstrap was used in this dissertation. To generate a moving block bootstrap realisation of the time series, select a block that is x

days in length from the time series across all stocks. This sample should be chosen randomly. If the original series had y days, then we choose y/k blocks to ensure that we have enough blocks to obtain a series with approximately the same length as the original series. The sampling is done with replacement as the means and correlations of the data is preserved [19]

Since the length of the bootstrapping period is denoted by x, a decrease in the length of the bootstrapping period to generate the data corresponds to shortening of the "memory" in the time series. If the bootstrapping length is set to $x = 10$ days, then the "memory" in the data cannot be longer than 10 days. When the length is set to $x = 1$ day, the process has no memory, each t^{th} realisation can be considered independent of the $(t-1)^{th}$ realisation. Therefore, given enough historical data, the bootstrapping technique can be used to test the effect of "memory" on the proposed investment strategy.

Chapter 5

Data and Results

This section begins with a presentation of the results from principal component analysis, followed by a comparison of the various implementations of the residual process. Finally an overview of the OU model parameters and other key variables is presented.

5.1 Principal component analysis

The results from principal component analysis have been incorporated to demonstrate the applicability of the technique in studying stock markets. This was done by using various time series data which includes daily total return data for FTSE JSE Top 40 index, as well as that of the constituents of the FTSE JSE Top 40 index from December 2002 to December 2004 (this data was used for all the results in this sub-section except for where it was mentioned otherwise), returns of Dow Jones Index from April 1998 to December 2002, and values for the CBOE VIX from April 1998 to December 2012 were also used in this subsection.

The proportion of the variance in the data structure that is explained by PCA is given by equation 3.10. The chart below shows the proportion of variance explained by each of the principal components. The first component explains about 25% of the variance. Five principal components are required in order to explain 50% of the variance.

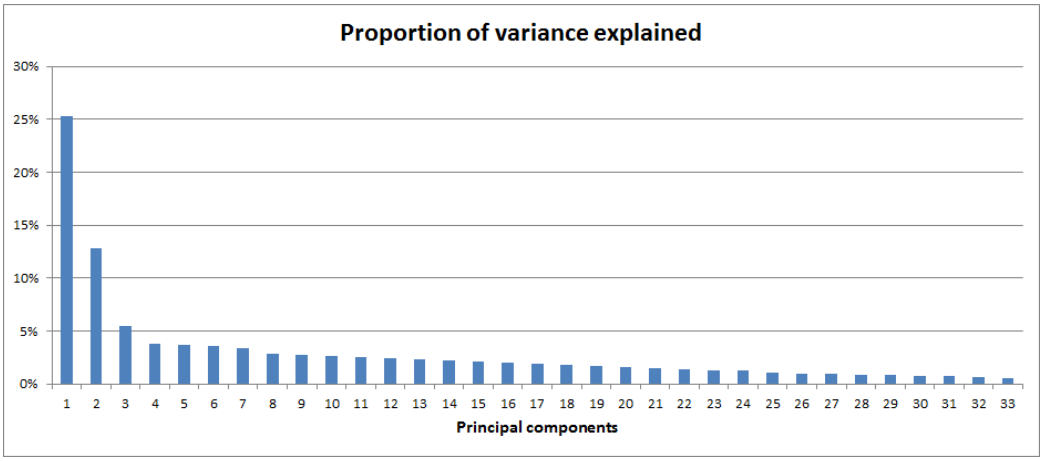

Figure 5.1: The first principal component

The dominant eigenvector consists of only positive weights for all the shares in the market being considered. It has been found by several authors [7] that the dominant eigenportfolio is analogous to the market portfolio in the Capital asset pricing model (CAPM) if the weights are adjusted for volatility. The chart below shows the make-up of the first eigenvector (also referred to as eigenportfolio in this dissertation).

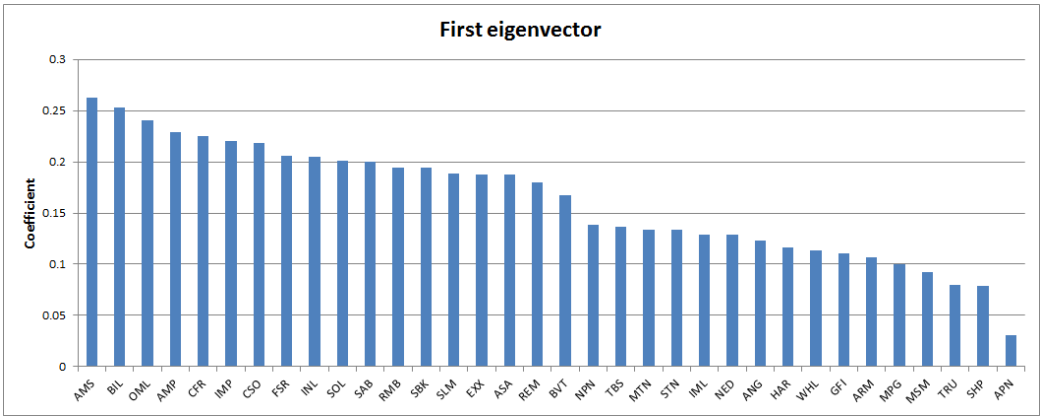

Figure 5.2: Loadings for the dominant eigenvector

The following chart shows a comparison of the evolution of the first principal component and that of the JSE Top 40 Index, which is a market capitalisation weighted portfolio (or the market portfolio in the language of the CAPM). The evolution of the first principal component is fairly similar to that of the

index. (Note that the return time series for the dominant eigenvector is given by the first principal component)

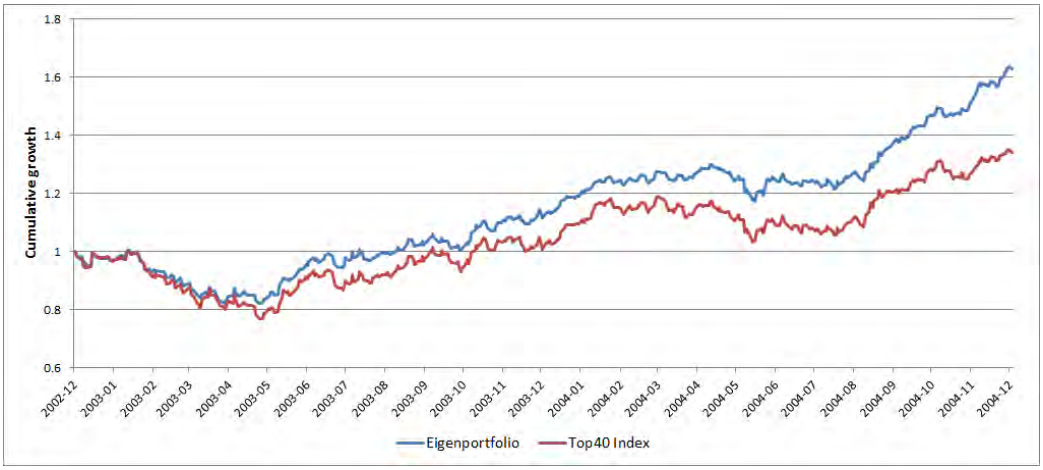

Figure 5.3: A comparison of the evolution of the dominant eigenportfolio and that of the market capitalisation weighted portfolio of stocks in the JSE Top40 Index

A regression analysis of the returns of the first eigenportfolio and the index was also performed. The regression had an R^2 of 88%, which shows that the first eigenportfolio is highly correlated with the index. The results of the regression, which are tabulated below, confirm the strength of the relationship between the first eigenportfolio and the market capitalisation weighted index.

Table 5.1: ANOVA table for the regression of the first eigenportfolio and the market capitalisation weighted index

ANOVA						
	df	*SS*	*MS*	*F*	*Significance F*	
Regression	1	0.060	0.060	3905.324	1.7344E-244	
Residual	522	0.008	0.000			
Total	523	0.068				
	Coefficients	*Standard Error*	*t Stat*	*P-value*	*Lower 95%*	*Upper 95%*
Intercept	-0.001	0.000	-3.309	0.00100	-0.000906848	-0.00023114
X Variable 1	1.223	0.020	62.493	0.00000	1.184576879	1.26147091

The second eigenvector explains 13 % of the variance. The coefficients for the second eigenvector display significant clustering within market sectors

when arranged in descending order (which is a property that is referred to as coherence). For example financials (shown as green bars in figure 5.4) had large positive coefficients, whilst resource shares (red bars) had large negative coefficients and industrial stocks dominate the middle with coefficients ranging from small positive to small negative values.

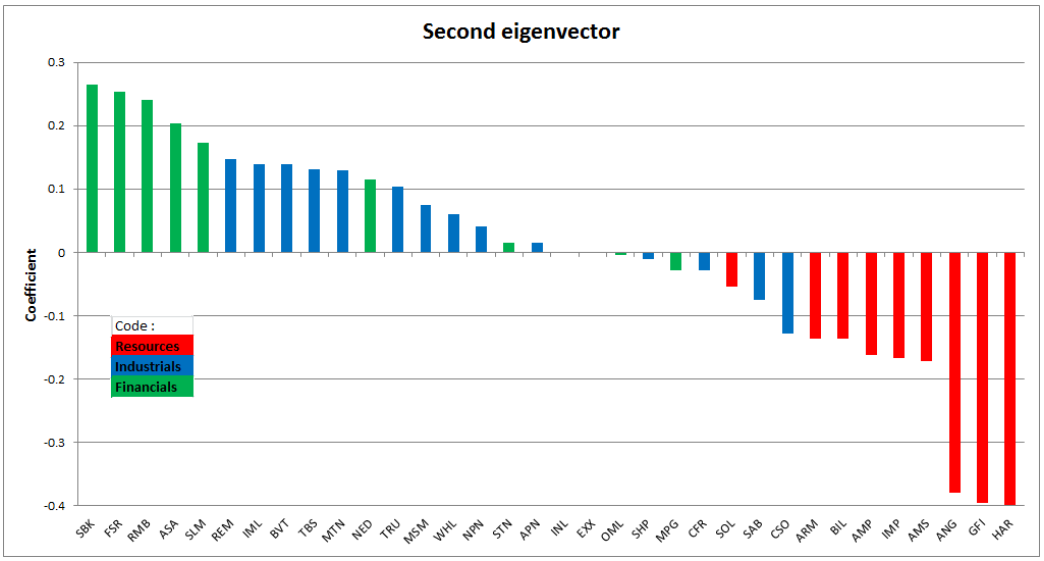

Figure 5.4: Principal components loadings of the second eigenvector exhibiting coherence

The historical return data for the constituents stocks of the Dow Jones index and the CBOE VIX index from 1998 to 2012 was used to plot figure 5.5 which shows that the number of principal components that are required to explain a set amount of the variance is not constant, it changes when the correlations change. For instance, during times of very high volatility, shares tend to be highly correlated and the number of components required to explain a set market variance is lower.

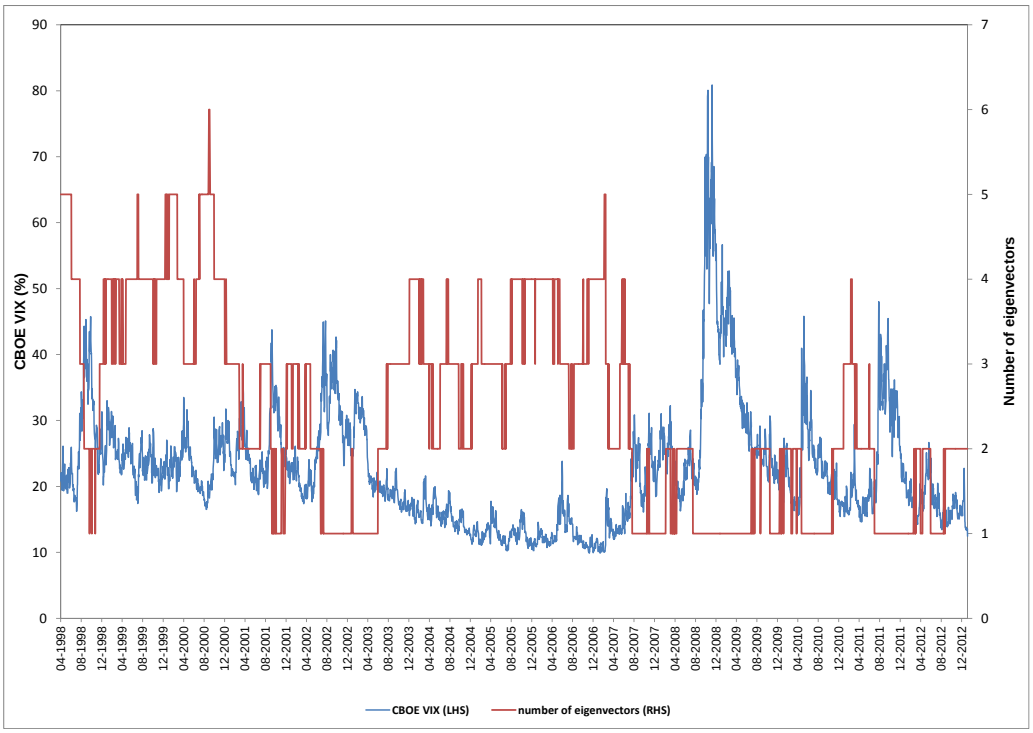

Figure 5.5: Inverse relationship between volatility and the number of principal components required to explain a set variance

The next chart (figure 5.6) shows how the proportion of the variance that is explained by the first principal component varies over time. (This chart uses the same data as figure 5.5 above).

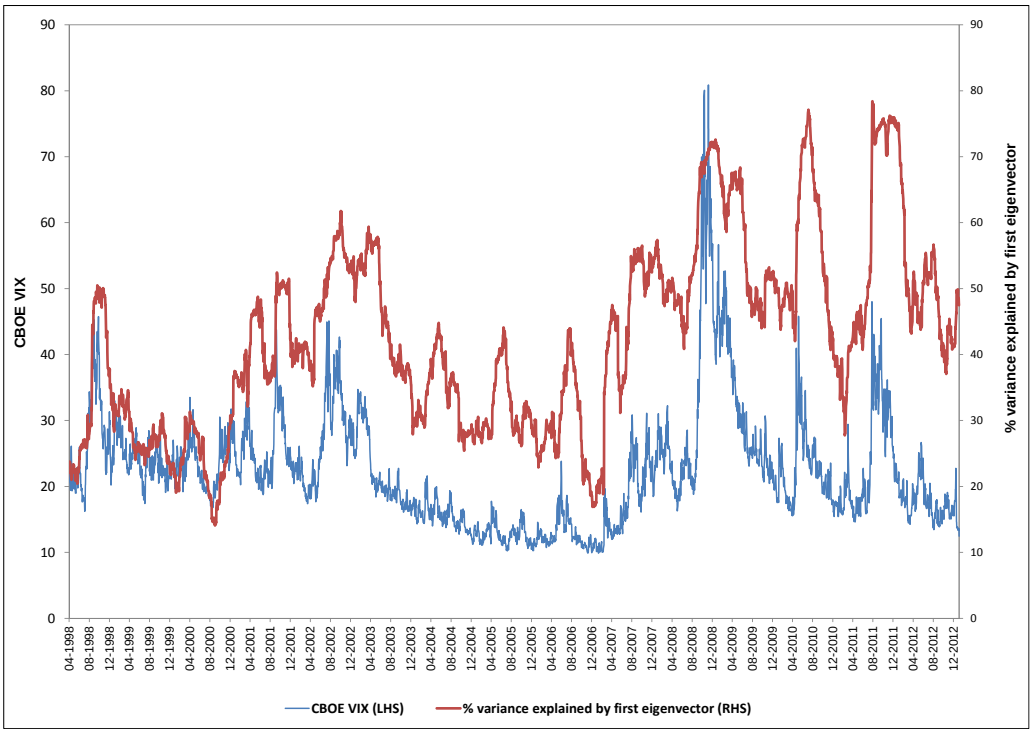

Figure 5.6: The proportion of the variance in the data structure that is explained by the first principal component

The amount of variance explained by the first principal component increases to over 50% during periods of extreme market volatility, decreasing to 15% when volatility is low. This result is complimentary to that of figure 5.5 above.

Finally figure 5.7 below shows that the number of principal components required to explain a set variance may change over time, depending on the prevailing market conditions.(JSE Top40 data was used for figure 5.7)

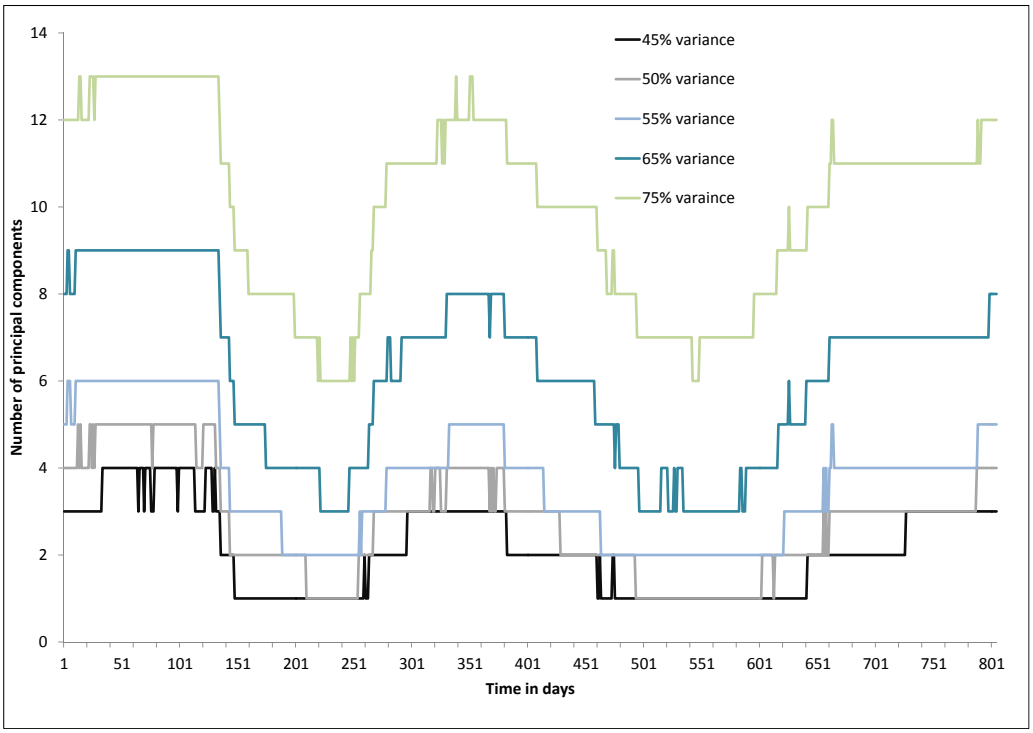

Figure 5.7: The number of PCA factors required for explaining different levels of set variance

5.2 Residual process

The purpose of this section is to evaluate the different implementations of the residual process model models through backtesting analysis (the models were discussed at length in subsection 4.2).

In contrast to the results in section 5.1 above, the selection of a model for implementing the residual process will have an impact on the results from backtesting the strategy. Therefore, the models' of the residuals were evaluated using an time series data that predates the data that is used for backtesting (see section 4.5 for more information on sample selection for backtesting)

The initial data sample that was chosen was the total return data for the constituents of FTSE JSE ALSI Top 40 index data for the two years before

December 2001. This data sample was used for determining the variables such as trading signals and training windows etc. (refer to section 4.3 for more information on which variables have to be set for the investment strategy). It follows from this that the strategy could evaluated on data from December 2001 to December 2013 (out of sample data)

The residual process models were evaluated using the trading signals that were used by [7],

$$\text{buy to open if } s_i < -1.25$$

$$\text{sell to open if } s_i > +1.25$$

$$\text{close long positions} s_i > -0.5$$

$$\text{close short positions} s_i < +0.75$$

In order to evaluate the models', the training window was set to half a year (approximately 132 tyrading days) as JSE listed companies are only required to report twice a year. This is in line with the work done by [7]. The table below shows a summary of the results

Table 5.2: Assessing the residual process' models

	Constant parameters	No replacement	With replacement
Annualised returns	12.03%	10.26%	13.95%
Total number of trades	168	210	250
Number of failed trades	50	40	35
Average trade length (days)	29.0	24.1	19.4

The results show that the residual model with constant parameters has a longer average trade length, 29 trading days versus 19 trading days for the model with updating of residuals and replacement of older residuals. The results also show that using constant parameters leads to fewer trades - this is in line with expectations, as the model has a slow alpha decay. The number of trades that trigger the stop loss (failed trades) is also higher when constant parameters are used.

Overall the results show that using dynamic OU parameters with updating and replacement is the best implementation with regards to returns and speed of mean reversion. The model using dynamic parameters without replacement is a compromise between the other two, however it is noted that

it faired the worst with regards to returns.

The chart below compares the cumulative performance of the three implementations over time.

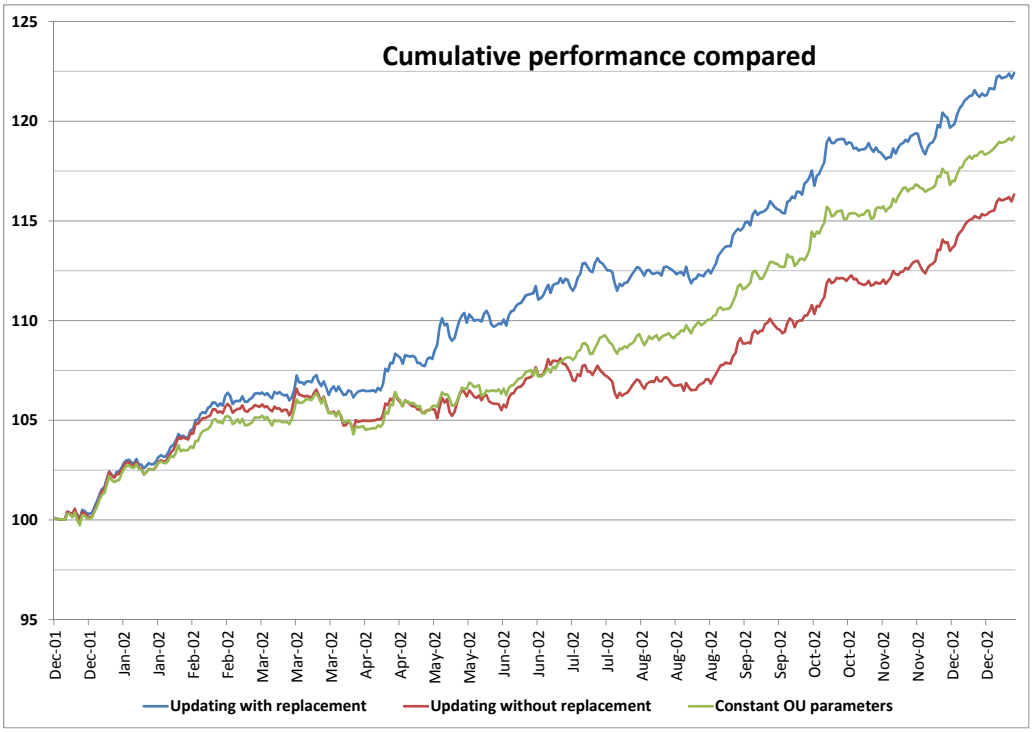

Figure 5.8: Comparison of the residual process models

5.3 Parameter estimation and model specification

Recall that the OU parameters have to be estimated on a rolling basis as such the results in sub-section 5.3.1 have been included for illustrative purposes. Section 5.3.2 details the results for specifying the remainder of the variables required for the investment strategy.

5.3.1 OU parameters

The table below provides the typical magnitudes of the OU parameters for the model, the parameters were estimated for illustration purposes for selected stocks which are listed below. Note that trades are not opened if the parameter b which is the speed of mean reversion is greater than 0.93. This is equivalent to about 15 days.

Table 5.3: Typical values for OU parameters

share	a	b	m	sigma	reversion days
Absa	0.00%	0.88	0.31%	2.22%	7.4
Anglos	-0.01%	0.93	0.33%	3.21%	13.3
Bidvest	-0.04%	0.93	0.18%	2.94%	12.86
Richemont	-0.04%	0.89	0.06%	3.42%	8.9
Investec	-0.03%	0.91	0.32%	3.89%	10.9
Harmony	-0.02%	0.92	0.52%	3.21%	12.4
MTN	0.07%	0.93	0.80%	4.72%	14.6
Standard bank	0.00%	0.91	0.43%	2.28%	10.8
Sasol	-0.07%	0.92	0.56%	3.96%	12.3
Truworths	-0.01%	0.88	0.22%	2.82%	7.4
Woolworths	0.11%	0.91	1.67%	3.46%	10.4

Figure 5.9 below shows a typical distribution of the speed of mean reversion for the shares in the investment universe.

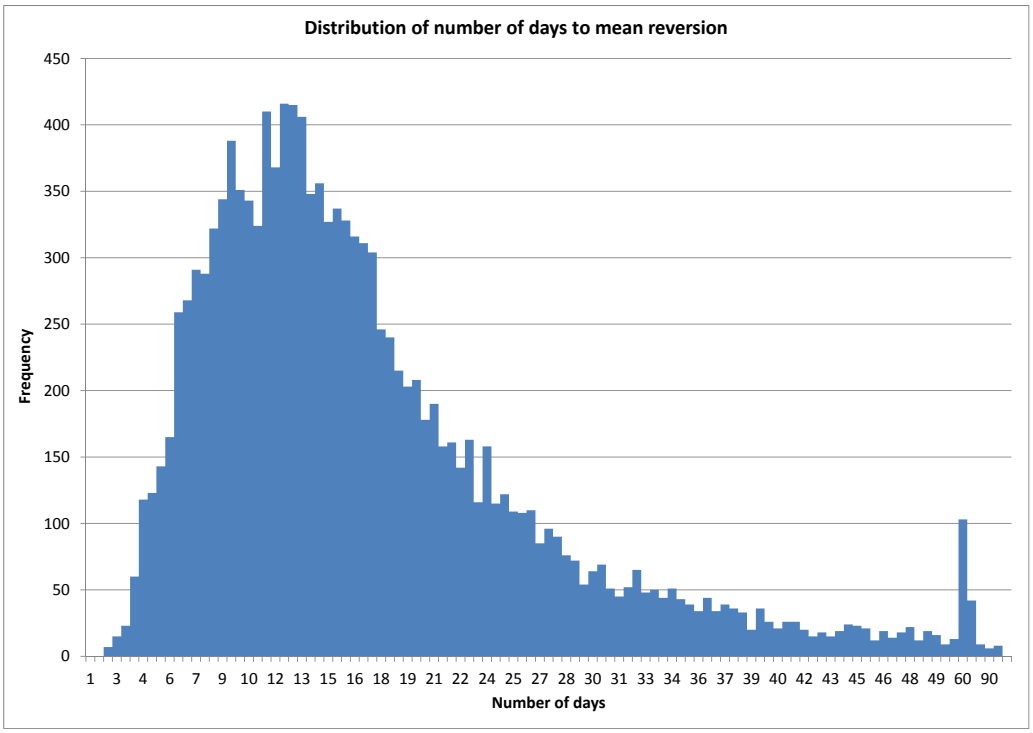

Figure 5.9: Typical distribution for the time it takes for mean reversion

The chart shows that the number of days to mean reversion for the model over time, it is skewed to the left (fewer days to reversion), which encourages trading as there are many mean reverting opportunities. Note the outlier when number of days equals 60, this is due to the fact that trades that remain open after 59 days are terminated. Furthermore note that there are data points beyond 60 days, this because not all subject stock i would have open trades at all times, so these are just some of the data points for "potential" trades that were never executed (the rule to terminate after 60 days can only be applied to open trades).

5.3.2 Training window and trading signal

As discussed in section 4.3, the optimal training window was determined based on simulations performed using historical data. Simulation experiments of the strategy were performed using data for stocks on the JSE TOP40 over the two years to 31 December 2001. The experiments were designed to determine the optimal training window for performing principal component

analysis as well as estimating the regression model used when estimating the OU model. In addition, the simulations were then used to determine the trading signals.

The table below shows the average annualised return corresponding to each training window size in the simulation experiment. The simulations were performed over a range of trading signals. Note that the same training window was used for both PCA and the regression process.

Table 5.4: Determing an optimal training window

Training window length (days)	90	120	150	180	210	240	270
Annualised returns (%)	15.98	11.21	12.15	13.59	19.21	31.32	18.4

The results show that a training window of 240 trading days would yield the highest return. The outcome of these simulations was then used to determine the optimal trading signals. The simulations were performed using the same data as above over a range of trading signals and the training window was set to 240 days. An average of the best cut-offs for opening and closing a position were used in the subsequent analysis. The optimal trading signals determined are as follows,

$$\text{buy to open if } s_i < -1.30$$

$$\text{sell to open if } s_i > +1.20$$

$$\text{close long position} s_i > -0.95$$

$$\text{close short position} s_i < +0.85$$

It was also found that the strategy performs better when trades are opened when the estimated time to mean reversion is shorter than 14 days. This corresponds to imposing the condition $b < 0.93$. Note that b is one of the OU model parameters, the lower the value of b, the faster the mean reversion.
’

5.4 Evaluating the strategy

The aim of this section is to assess the performance of the proposed strategy as a whole. To this end, the relevant results from the previous sections were used. The strategy was evaluated using backtesting experiments.

The performance of the strategy was measured using equation 4.4. The return data for the constituent stocks of the FTSE/JSE ALSI Top 40 Index and the FTSE/JSE ALSI Index from 19 December 2001 to 19 December 2013 was used.

5.4.1 Performance of the strategy

The table below shows the performance of the strategy. The performance of the model was compared to cash and the return of the FTSE/JSE ALSI Top 40 Index. Trading costs were initially set to 10 basis points (round trip costs). Note that all the figures in the tables have been annualised and are expressed as percentages. Also note that there was no leverage employed used when evaluating the strategy at this stage (in the language of equation 4.5, we can also say the leverage ratio was set to 100%). It is important to note that as the strategy was evaluated from 19 December 2001 to 19 December 2013, the year reference in the table does not exactly match the calendar year.

Table 5.5: Performance of the strategy

	Strategy	Top 40 Index	Long/Shorts	Cash
2013	-3.7	17.2	-6.6	5.7
2012	-3.1	26.4	-6.6	5.9
2011	15.9	1.9	11.1	6.8
2010	18.0	17.5	12.3	8.5
2009	35.8	35.1	26.1	13.0
2008	34.7	-27.9	25.8	10.9
2007	14.7	22.5	9.1	8.5
2006	7.7	40.6	2.7	7.6
2005	16.5	48.6	10.8	8.3
2004	17.8	23.7	9.4	11.7
2003	18.4	13.4	9.7	13.5
2002	48.6	-14.5	40.2	10.8

The annualised returns for the strategy are shown under the column labelled Strategy, the returns for the index are shown under the column labelled Top 40 Index, and the returns under the column long/shorts are the returns of the strategy excluding the interest rate earned on capital. Recall that the capital allocation decision given by equation 4.4 requires excess capital to be placed in a bank account to earn the cash rate because the strategy is a long-short strategy. (The long/short column in table 5.5 is an attempt to provide an attribution for the strategy's performance that is due to the long/short trades).

The results show that the returns for the strategy on any given year can be roughly approximated as the sum of returns from long/short trades and the interest earned from a cash deposit. Overall the strategy returned 17.28% (annualised for the period) whilst the index returned 15.62% over this period. The return from cash deposits over the was 8.96%. It is also worth noting that the The performance of strategy was especially poor over the last two years (2012 and 2013). This is possibly because the trading signals became somewhat stale. As discussed previously, the trading signals were determined once using data from 2000 to 2001.

The chart below shows the cumulative performance of the strategy compared to that of the index and cash over time.

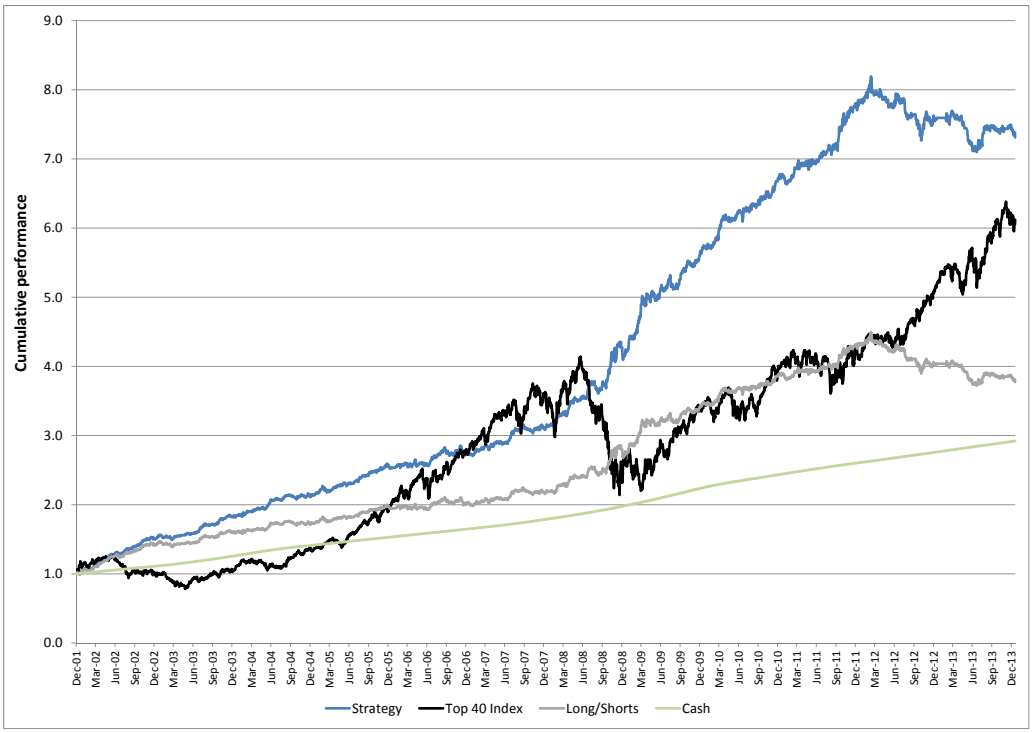

Figure 5.10: Cumulative performance of the strategy

The annualised out-performance of the strategy may seem marginal but its more significant when one considers the volatility of the returns from the strategy versus that of the index. In the table below we note that the annualised volatility of the strategy is less than that of the index which would suggest that the risk-adjusted returns of the strategy are much higher. This would imply that one could employ leverage to earn much larger profits (than that of the index) for an equivalent level of risk.

Table 5.6: Volatility of the strategy and the index

	Strategy	Top 40 Index	Long/Shorts
2013	6.1	12.4	6.1
2012	7.5	20.7	7.5
2011	7.7	18.7	7.7
2010	6.9	25.3	6.9
2009	9.7	39.6	9.7
2008	12.1	22.3	12.1
2007	7.2	23.8	7.2
2006	8.6	14.8	8.6
2005	7.0	15.8	7.0
2004	7.9	19.5	7.9
2003	6.6	20.2	6.6
2002	9.6	23.6	9.6

Note that all of the strategy's volatility is due to the long/short component of the returns. This is because the volatility of cash is nearly zero. These finding may also be interpreted with the aid of the Sharpe ratio which are showin in table 5.7.

Table 5.7: Sharpe ratios of the strategy versus the index

	Strategy	Top 40 Index
2013	-1.54	0.93
2012	-1.20	0.99
2011	1.17	-0.26
2010	1.38	0.35
2009	2.34	0.56
2008	1.98	-1.74
2007	0.86	0.59
2006	0.01	2.22
2005	1.17	2.56
2004	0.77	0.62
2003	0.74	-0.00
2002	3.92	-1.07

The average annual Sharpe ratio over the period was 0.96 which compares favourably to the Sharpe ratio of 0.48 attained by the index. This means that the strategy is able to obtain a higher risk adjusted return than that of the index.

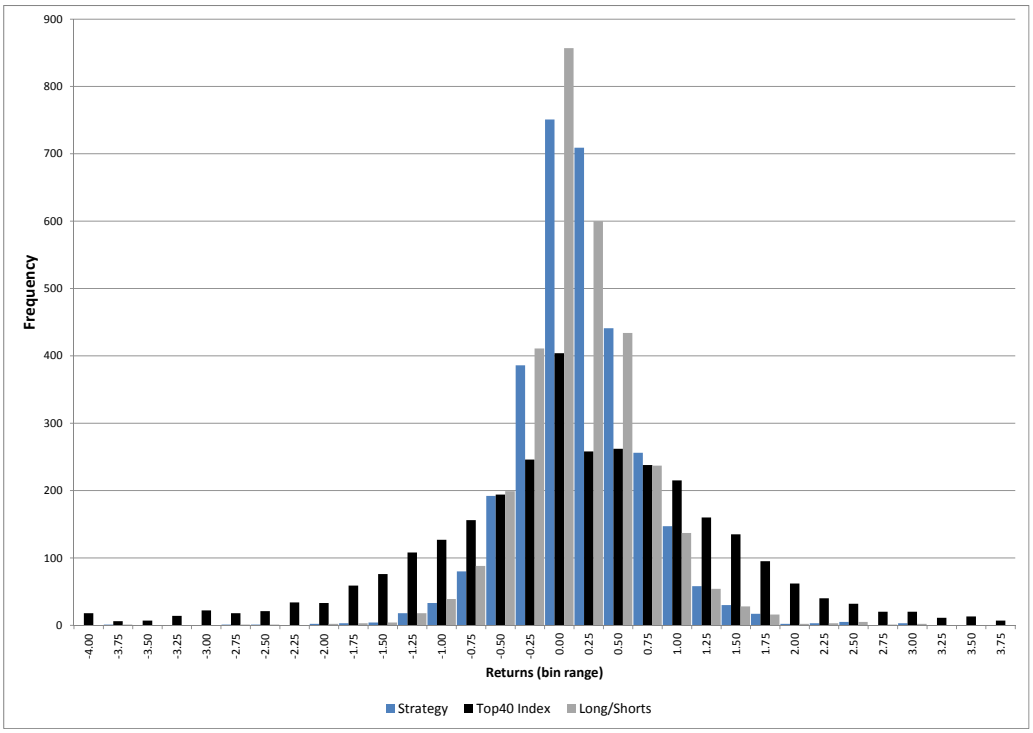

Figure 5.11: Distribution of the strategy's daily returns

In addition, the histogram in figure 5.11 can also be used to reiterate the low risk nature of the strategy. Over 96 % of the strategy's daily returns fall between -1.25% and 1.25% whilst only 70% of the index's daily returns fall within that range. This also confirms that the strategy's daily performance is very consistent.

5.4.2 Impact of trading costs

Trades were simulated for different levels of trading costs. The table below shows the performance of the strategy under different trading cost assumptions (10 bps and 20 bps).

Table 5.8: Impact of trading costs on returns

	10 bps	20 bps
2013	-3.7	-5.3
2012	-3.1	-5.2
2011	15.9	12.3
2010	18.0	14.2
2009	35.8	31.2
2008	34.7	29.1
2007	14.7	11.4
2006	7.7	3.8
2005	16.5	12.1
2004	17.8	14.3
2003	18.4	14.0
2002	48.6	43.0

Trading costs have a significant impact on the returns that investors earn from the strategy. As a result, these types of strategies are more suitable to agents who are able to transact at reasonable costs. Nevertheless, the impact (of doubling trading costs to 20 bps) is not unbearable as investors are still able to outperform cash even when the round trip costs are 20 bps. For s long as investors can outperform cash at reasonable levels of risk, leverage can be employed to enhance return for as long as the strategy (without leverage) is able to outperform cash at reasonable levels of risk.

5.4.3 Impact of increasing leverage

Trades were simulated for different levels of leverage, whilst keeping round trip costs at 10 bps. The impact of increasing leverage from 100% to 150% and 200% is illustrated on the following table.

Table 5.9: Impact of leverage on returns

	100%	150%	200%	Cash
2013	-3.7	-6.2	-8.8	5.7
2012	-3.1	-5.5	-7.9	5.9
2011	15.9	23.7	32.0	6.8
2010	18.0	26.6	35.8	8.5
2009	35.8	54.3	74.9	13.0
2008	34.7	53.6	74.5	10.9
2007	14.7	21.1	27.7	8.5
2006	7.7	10.9	13.9	7.6
2005	16.5	24.6	33.1	8.3
2004	17.8	24.9	32.2	11.7
2003	18.4	25.7	33.4	13.5
2002	48.6	78.5	113.9	10.8

The results show that the performance of the strategy can be enhanced by increasing leverage. This can be done whilst keeping volatility at a manageable level because the strategy has a low risk inherent characteristic. As a result, investors are able to somewhat calibrate the strategy to their preferred risk tolerance. The annualised return over the period was 25.3% when leverage was set to 150% and 33.5 % when leverage was set to 200%. This compares favourably with the 17.3% that the strategy earns without leverage. The table below shows the impact that leverage has on the strategy's volatility.

Table 5.10: Impact of leverage on volatility

	100%	150%	200%
2013	6.1	9.1	12.2
2012	7.5	11.2	14.9
2011	7.7	11.6	15.4
2010	6.9	10.3	13.7
2009	9.7	14.6	19.5
2008	12.1	18.1	24.1
2007	7.2	10.7	14.3
2006	8.6	13.0	17.3
2005	7.0	10.4	13.9
2004	7.9	11.8	15.7
2003	6.6	9.9	13.2
2002	9.6	14.4	19.3

Hence, at anything below 200% leverage, the strategy maintained a level of

volatility that is below that of the Index.

5.4.4 Impact of the size of the investment universe

The results so far were for trade simulations performed using the data for the constituent stocks of the Top 40 index. This index consists of about 40 of the largest capitalised stocks on the JSE. In order to understand the impact of a larger opportunity set, trade simulations were performed over the same period, using the same trading signals and training window as above but using constituent data for the FTSE/JSE ALSI Index (109 stocks were used for this simulation as opposed to only 40 stocks in Top 40 index). The table below shows the performance of the strategy for this investment universe when to compared the universe consisting of stocks in the Top 40 index.

Table 5.11: Returns for the strategy for different sizes of the investment universe

	Larger universe (ALSI)	Top 40 Index universe	Cash
2013	11.2	-3.7	5.7
2012	8.6	-3.1	5.9
2011	14.4	15.9	6.8
2010	18.2	18.0	8.5
2009	18.0	35.8	13.0
2008	28.8	34.7	10.9
2007	18.5	14.7	8.5
2006	22.3	7.7	7.6
2005	6.9	16.5	8.3
2004	8.5	17.8	11.7
2003	33.5	18.4	13.5
2002	36.9	48.6	10.8

The results show that using a broader opportunity set improves performance. This is not surprising since the larger the investment universe, the larger the number of trades that can be initiated. When simulations were performed using the shares in the ALSI Top 40, 894 trades were executed between 2002 and 2013; of which only 37 trades hit the stop loss. For the broader JSE All Share Index, 2461 trades were executed; of which 140 hit the stop loss.

The chart below shows the cumulative performance when using the broader investment universe, this is contrasted to the performance of the strategy

when the JSE Top 40 was the investment universe. Note that long/short in the chart refers to the long/shorts when the investment universe is the constituents of the broader ALSI index and not the Top 40 index

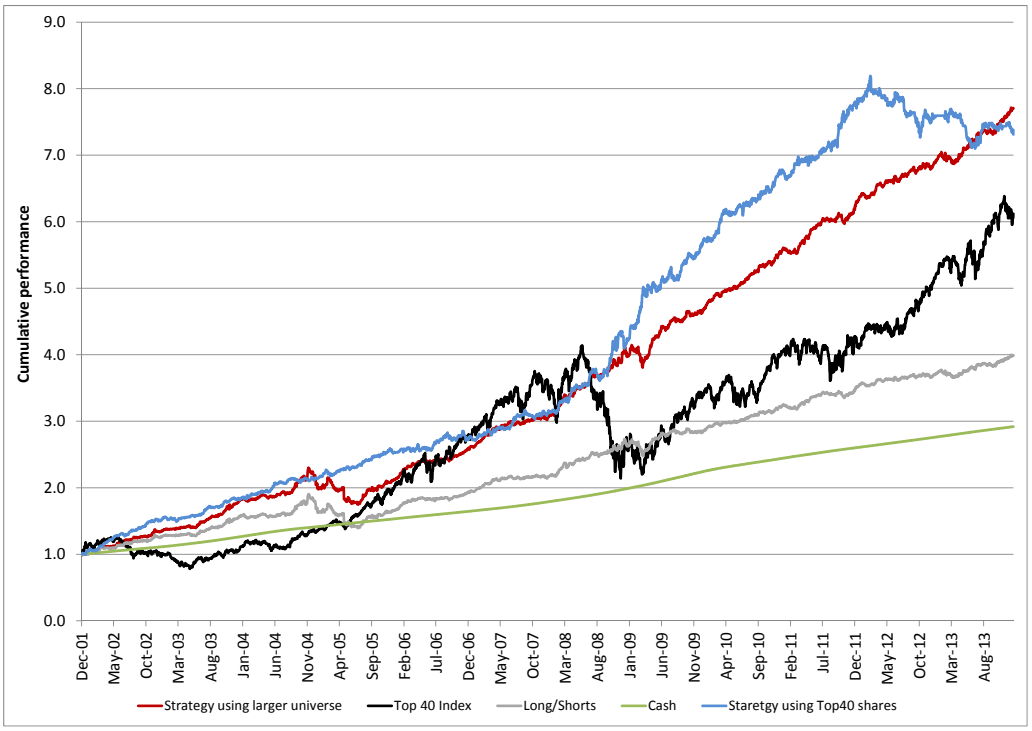

Figure 5.12: Cumulative performance of the strategy using larger investment universe

In addition the table below shows that the volatility of the strategy is lower when a broader investment universe is used.

Table 5.12: Volatility of the strategy for different sizes of the investment universe

	Larger universe (ALSI)	Top 40 Index universe
2013	4.7	6.1
2012	3.8	7.5
2011	4.2	7.7
2010	4.0	6.9
2009	7.6	9.7
2008	8.6	12.1
2007	5.1	7.2
2006	5.4	8.6
2005	14.3	7.0
2004	11.9	7.9
2003	9.3	6.6
2002	8.4	9.6

In recent times the volatility for using the broader investment universe was generally lower when the investment strategy was employed over the JSE Top40 Index. Note that in reality the cost of trading shares in the broader All Share Index would be on average higher, as the some of the stocks have poor liquidity when compared to the stocks in the Top 40 Index.

5.4.5 Monte-Carlo Simulations

In order to simulate stock returns using the Monte Carlo technique, the drift vector $\vec{\mu}$ and the covariance matrices $\vec{\sigma}$ have to be specified. The mean vector and covariance matrix used in the simulations were the mean vector and the covariance matrix of the same historical time series data used for backtesting (i.e. data for the constituent stocks of the JSE ALSI Top40 Index from 2001 to 2013). The Monte Carlo simulations were performed 50 times, so that 50 data samples were generated. Trading costs were set to 10 bps (round trip)

The average return (annualised) for the strategy when implemented over Monte Carlo generated data was -0.75%, the median return was 1.02% the strategy generated negative returns 64% of the time. Trading costs detracted about 1.48% (annualised). It is important to note that the return from cash has been stripped out in order to show the performance of the long-short trades.

The chart below shows the distribution of the returns from applying the strategy over data generated from the

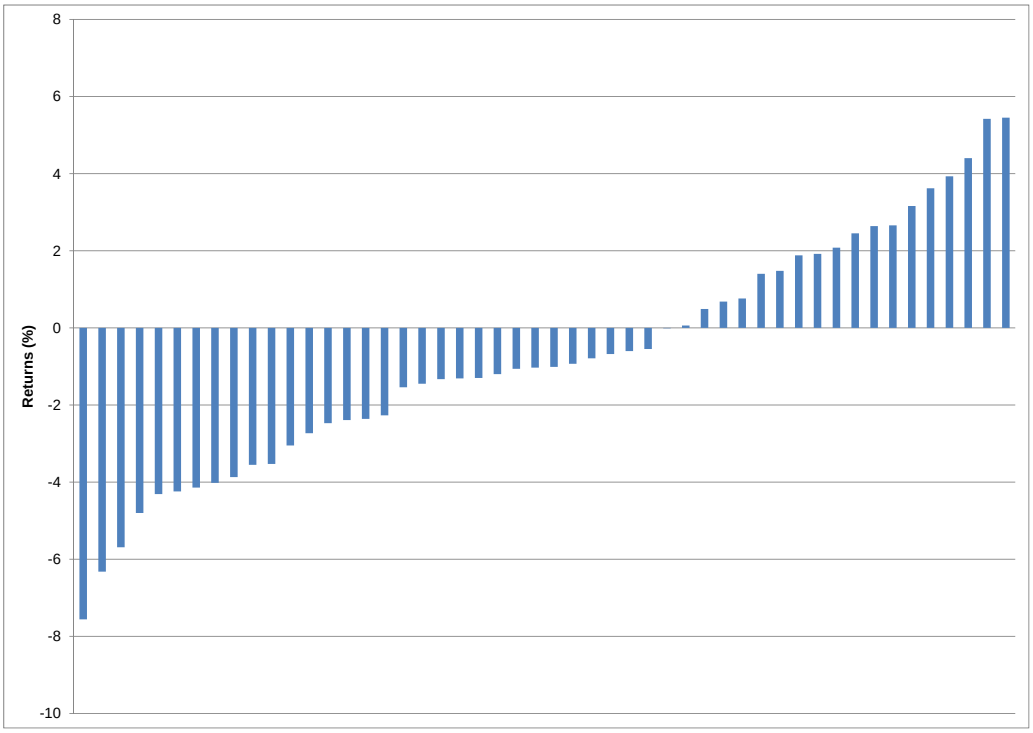

Figure 5.13: Distribution of the returns from applying the strategy over Monte Carlo generated data

The results from implementing the strategy over data generated from Monte Carlo simulations show that the strategy is sub-optimal in a world with independent increments. This means that as the market becomes asymptotically efficient, the strategy under-performs cash. The presence of transaction costs and other market frictions would make this strategy yield negative returns on average, if the market is just sufficiently efficient, so that the returns from the strategy are not enough to recoup costs.

5.4.6 Bootstrapping

In this section the model is evaluated on bootstrapped historical data. The historical data used for bootstrapping is the returns for the constituent stocks of the FTSE JSE Top40 Index from 2001 to 2013. The bootstrapping length

x was set to start at 1, increasing to 60. The average return at each set value of x was then recorded. It is important to note that this is the return of the long/short trades, so it excludes the impact of cash. The graph below shows how the profitability of the strategy is impacted by the bootstrapping length.

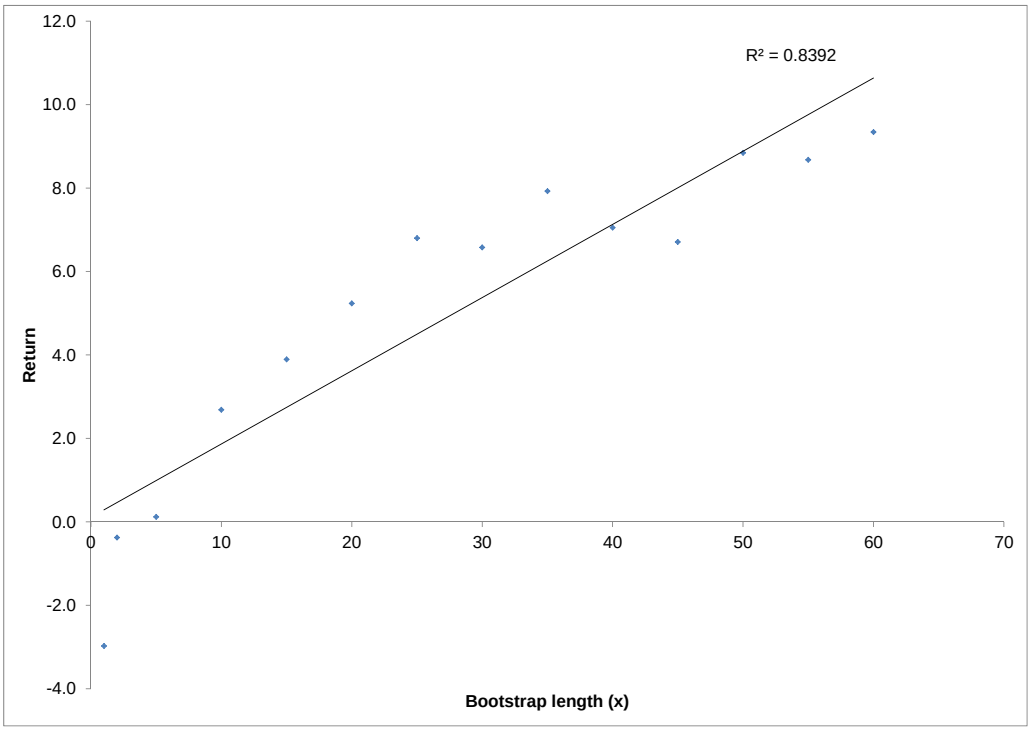

Figure 5.14: Performance of the strategy improves with an increase in bootstrap length

The performance of the strategy in negative at very low bootstrapping lengths ($x = 1$ and $x = 2$). When $x = 1$, the data that is generated by bootstrapping has independent increments, so the performance of the strategy is expected to not be to be poor[1] The performance improves in a linear fashion with an increase in the bootstrap length.

The results seem to suggest that the daily increments in the prices of the stocks which were under consideration were somewhat dependent on the previous realisations (i.e the increments are not sufficiently independent).

[1]The results from Monte Carlo simulations show that the strategy performs poorly in market with independent increments.

Chapter 6

Conclusions

The purpose of this dissertation was to use computational tools to develop a trading strategy that exploits the potential of temporary mis-pricings in security markets. The strategy implemented in this dissertation was based on the work done by [7] which introduces the idea of using PCA and the OU model for generating trade signals.

The results of the thesis suggested that it is possible to find patterns in financial markets that may be exploited with a computational trading strategy. The strategy outlined in this dissertation was able to earn returns above those earned from investing in cash over most of periods for which it was evaluated. Furthermore the strategy was able to earn significant returns that are above those of the market index without taking on more risk or employing leverage. Additionally since the strategy is inherently less risky than investments in the index, its performance can be enhanced by using leverage.

The performance of the strategy for the period from 2012 to 2013 was poor when compared to the index. In particular, the returns from implementing the strategy over the universe of the JSE Top 40 stocks were negative. This is possibly due to the stale trading signals (recall that trading signals were determined using data from the year 2000 to the year 2001). However, it is unlikely that this is the only cause. Given this, further research on the causes of the significant under-performance over the past two years is required.

When the strategy was evaluated over data generated by Monte Carlo simulation, its performance was very poor. The strategy is not relevant in a world where stock prices exhibit independent increments. It is important to note that this is in line with expectations.

The results from evaluating the strategy over data generated from bootstrapping seem to suggest the existence of "memory" in the evolution of prices for stocks in the JSE Top40 Index.

Chapter 7

Recommendations

The dissertation was focused on implementing the statistical arbitrage strategy as proposed by [7]. However, there are many avenues for improving the strategy. We have listed below some of the recommendations.

- Genetic algorithms may be used in place of linear regression to find generate trades that have a higher degree of mean reversion.
- Consider the use of other mean reverting stochastic processes in place of the Ornstein-Uhlenbeck process. In order to do this, one can start with a discrete-time series model (such as the moving average auto-regressive (ARMA)) and then to find an equivalent continuous-time model. [20] has translated various discrete-time time series models to continuous-time equivalents.
- Investigate whether the performance of the strategy is overly reliant on stock market volatility. This might also shed a light on the causes of the poor performance between 2012 and 2013.
- Investigate the viability of placing some of the excess cash in an equity investment (such as an equity index) instead of money market deposit.
- Determine the optimal trading signals on a rolling basis to ensure they reflect the most recent market dynamics

Bibliography

[1] A.W Lo and J Hasanhodzic. *The Evolution of Technical Analysis: Financial Prediction from Babylonian Tablets to Bloomberg Terminals*. New York: John Wiley and Sons, 2011.

[2] Fama EF. Efficient capital markets: A review of theory and empirical work. *THe Journal of Finance*, 1970.

[3] G.B Malkiel. *A Random Walk Down Wall Street*. New York: W. W. Norton and Co., 1973.

[4] A.W Lo and C MacKinlay. *A Non-Random Walk Down Wall Street*. Princeton: Princeton University Press., 2010.

[5] M.H Pesaran. Predictability of asset returns and the efficient market hypothesis. *Cambridge Working Papers in Economics*, 2010.

[6] A.N Burgess. *A Computational Methodology for Modelling the Dynamics of Statistical Arbitrage*. PhD thesis, University of London : Department of Decision Sciences, 1999.

[7] Avellaneda M. and Lee JH. Statistical arbitrage in the us equities market. *Quantitative Finance*, 10:761 – 782, 2010.

[8] Hardie I. Beunza D and MacKenzie D. A price is a social thing: Towards a material sociology of arbitrage. 2006.

[9] Aswath Damodaran. Too good to be true: The dream of pure arbitrage. *http://people.stern.nyu.edu/adamodar/pdfiles/invphiloh/arbitrage.pdf/*, 2012.

[10] S.E Shreve. *Stochastic Calculus for Finance II*. Springer, 2004.

[11] T Bjork. *Arbitrage theory in continuos time*. Oxford University Press, 2003.

[12] B Oksendal. *Stochastic Differential Equations : An Introduction with Applications*. Oxford University Press, sixth edition, 2003.

[13] J.C Hull. *Options, Futures and Other Derivatives*. Prentice-Hall, fifth edition, 2003.

[14] O. Bondarenko. Statistical arbitrage and securities prices. *The Review of Financial Studies*, 16:875 – 919, 2003.

[15] A Pole. *Statistical Arbitrage: Algorithmic Trading Insights and Techniques*. John Wiley and Sons, Inc., 2007.

[16] R.A Johnson and W.D Wichern. *Applied Multivariate Statistical Analysis*. Sixth edition.

[17] N.S. Thomaidis, N Kondakis, and G.D. Dounias. An intelligent statistical arbitrage trading system. In *Advances in Artificial Intelligence*, volume 3955, pages 596–599. Springer Berlin Heidelberg, 2006.

[18] N Garleanu and L.H Pedersen. Dynamic trading with predicatable returns and transaction costs. *The Journal of Finance*, 68.

[19] B Efron and R.J Tibshirani. *An Introduction to the Bootstrap*. Chapman and Hall: New York, 1993.

[20] Cochrane J.H. Continuous-time linear models. 2012.

Appendix

Appendix A : Estimating the OU model

The OU parameters are estimated from equations 3.15 and 3.23. Assuming that $\Delta t = 252$. The three equations below are used to solve for the OU parameters

(1) $a = m(1 - \exp^{-\kappa\Delta t})$

(2) $b = e^{-\kappa\Delta t}$

(3) Variance$(\zeta) = \sigma^2 \dfrac{1 - \exp^{-2\kappa\Delta t}}{2\kappa}$,

Fitting the AR(1) model given by equation 3.23 it follows that

$\hat{X}_{i,j+1} = \hat{a} + \hat{b}X_{ij}$ and
$\zeta_{i,j+1} = X_{i,j+1} - \hat{X}_{i,j+1}$ now

$\hat{b} = e^{-252\hat{\kappa}}$ and
$\hat{\kappa} = -\mathbf{log}(\hat{b}) \times 252$

$\hat{\text{var}}(\zeta) = \hat{\sigma}^2 \dfrac{1 - \exp^{-2\kappa\Delta t}}{2\kappa} = \hat{\sigma}^2\big(\dfrac{1 - \hat{b}^2}{2\hat{\kappa}}\big)$

$\sigma = \sqrt{\dfrac{\text{Variance}(\zeta) \cdot 2\kappa}{1 - b^2}}$

Similar to reference [7], the equilibrium variance from equitaio 3.10 is used, so that

$$\sigma_{\text{eq}} = \sqrt{\frac{\text{Variance}(\zeta)}{1 - b^2}}$$

Appendix B : MATLAB code

This section following pages contain the MATLAB code that was used in the dissertation.[1]

[1]The functions ols and armaxfilter use in the dissertation were obtained from the MFE MATLAB Toolbox (c)2001-2009 Kevin Sheppard

```
[p,q] = size(data);
values

for last_entry = start : final

% set data
first_entry = last_entry-nrows+1;
pca_last = last_entry;
pca_first = last_entry - pca_rows + 1;
day = day + 1;
market_move = transpose(data(last_entry+1,:)); % time t+1 market movements used for↙
PnL calculation
dataM = data(first_entry:last_entry,1:q);
data_pca = data(pca_first:pca_last,1:q);

pca2 % run PCA module


[day_no,comp_num] = size(Vm); %record the number of components for set variance
weight = zeros(q,comp_num);
weight2 = zeros(q,comp_num);
beta_weighted = zeros(q,comp_num);
comp_rec(day) = comp_num;
share_weight = zeros(1,q);

for j = 1:comp_num;
for i = 1:q
if Rsigma(i) == 0
Rsigma(i) = 0.0204*2;
end
weight(i,j) = Vm(i,j)/Rsigma(i);
end
end

for j =1:comp_num  ;
for i = 1:q
weight2(i,j) = weight(i,j)/sum(weight(1:q,j));
end
end
eigenportfolio = dataM*weight2; % compute the historical returns for the↙
Eigenportfolios
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
reg3 % run the regresion and ar process module
model3 %trading run the trading module
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


for k = 1:q


for j = 1:comp_num;
for i = 1:q
beta_weighted(i,j) = beta(k,j)*weight2(i,j);
```

```
end
end

for i = 1:q
share_weight(i) = sum( beta_weighted(i,:));
end

if status(k)== 0 || day == 1
  eigen1(:,k) = share_weight;
end

% find portfolio holdings by using the regression output and the trading
% output
eigen_holdings(:,k) = eigen1(:,k)*(-status(k));

end

% portfilio holdings
for k = 1:q
    for p = 1:q
        if p == k
            trade_holdings(p,k) = eigen_holdings(p,k)+status(k);
        else
            trade_holdings(p,k) =  eigen_holdings(p,k)  ;
        end
    end
end


prev_trade = trade_holdings;


for k = 1:q
pnl_trade(k) = dot(trade_holdings(:,k),market_move); %compute PnL for modelling the↙
residual process
fund_trade(k) = sum(trade_holdings(:,k));
pnl_trade(k) = pnl_trade(k);
pnl_cum(k) = pnl_trade(k)+pnl_cum(k);
pnl_trade2(k,day) = pnl_trade(k);
if s(k) < 0
residual_process(k) = residual_process(k) + pnl_trade(k)-alpha(k) ;
%updating the residuals process with PnL
end
if s(k) > 0
residual_process(k) = residual_process(k)- pnl_trade(k)-alpha(k) ;
%updating the residuals process with PnL
end
% for debugging %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
if trade_length(k,day)~= 0
trade_earn(k,day) = sum(pnl_trade2(k,day-trade_length(k,day)+1:day-1));
cost_incurred(k,day) = sum(cost_trade(k,day-trade_length(k,day)+1:day-1));
trade_income(k,day) = trade_earn(k,day)-cost_incurred(k,day) ;
end

% creating the residuals process after a trade is open
```

```
if trade_open(k,day)> 2
current_earn(k,day) = sum(pnl_trade2(k,day-trade_open(k,day)+1:day-1));
end
if trade_open(k,day)~= 0
    r3(nrows+trade_open(k,day),k) = residual_process(k);
end
end

%%%%%%%%%%%%%%%%%%%%%%%%%
% trade unitisation
% conrolling leverage
for k = 1:q
num_trades = sum(abs(status));
if num_trades ==0
    num_trades=1;
end
trade_holdings(:,k) = trade_holdings(:,k)./(num_trades);
if status(k)~=0
    trade_holdings(:,k) = trade_holdings(:,k)./norm(trade_holdings(:,k));
    leverage_factor(k) = sum(abs(trade_holdings(:,k)));
trade_holdings(:,k) = trade_holdings(:,k)./leverage_factor(k);
end
end
for u = 1:q
port_pos(u) = sum(trade_holdings(u,:));
end
lever2(day) = sum(abs(port_pos));
if lever2(day)==0
    lever2(day) = 1;
end
for u = 1:q
port_pos(u) = 1*port_pos(u)/lever2(day);
end
net_position(day) = sum(port_pos);
actual_leverage(day) = sum(abs(port_pos));
% trade unitisation
% conrolling leverage
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%



pnl_actual = dot(port_pos,market_move); %compute actual PnL after controlling
% for leverage and unitising the trades

%%%%%%%%%%%%%%%%%%%%%%%%
profit_trade = zeros(g);
for k = 1:q
pnl_trade(k) = dot(trade_holdings(:,k),market_move)/lever2(day); % compute actual PnL↙
for each trade
fund_trade(k) = sum(trade_holdings(:,k));
pnl_cum(k) = pnl_trade(k)+pnl_cum(k);
pnl_trade_lever(k,day) = pnl_trade(k);
earnings(k,day) = pnl_trade(k);
end
```

```
revenue(day,1) = pnl_actual;
earnings2(day,1) =sum(pnl_trade);
for k = 1:q
book_weight(k) = sum(trade_holdings(k,:))/lever2(day);
end


book_change = abs(transpose(book_weight) - book_weightPrev);
cost = sum(book_change)*5/10000; % round trip costs at 10bps
trade_cost1(day) = cost;
cost1 = cost1 + cost;
book_weightPrev = transpose(book_weight);
if status == 0
    profit = 0;
    trade_cost = 0;
else

funds_rate = ((1+int_rate(day)/100)^(1/365)-1);
Eq0 = Eq;
% capital growth, equation 4.4 in report
Eq = Eq + Eq*(1-net_position(day))*funds_rate+Eq*revenue(day,1)-Eq*cost;
trade_cost = cost;
profit = Eq/Eq0-1;
end

returns(day,1) = Eq;
profit_earned(day) = profit;

%records
if day ~= 1
    cum_pnl(day) = cum_pnl(day-1)*(1+profit);
    cum_cost(day) = cum_cost(day-1)*(1+trade_cost);
    cum_rate(day) = cum_rate(day-1)*(1+int_rate(day)/day_frac);
    cum_index(day) = cum_index(day-1)*(1+index40(last_entry+1));
else
   cum_pnl(day) = 1+profit;
   cum_cost(day) = 1+trade_cost;
   cum_rate(day) = (1+int_rate(day)/day_frac);
   cum_index(day) = 1+index40(day);
end

score_evol(:,day) =  s;
status_evol(:,day) =  status;
%trade_evol(:,day) = b_param.*abs(status);

end

profit_vol = std(returns)*sqrt(252)*100;
% trade_count
% trade_fail
cum_return = cum_pnl(day);
annualised_return = (Eq^(252/day)-1)*100;
annualised_cost = (cum_cost(day)^(252/day)-1)*100;
annualised_rate = (cum_rate(day)^(252/day)-1)*100;
```

```
%sharpe_ratio = (annualised_return - annualised_rate)/profit_vol;
annualised_return;
ave_trade_length = sum(sum(trade_length))/trade_count;
trade_count;
trade_fail;
```

```
%regression and autoregression
beta1 = zeros(q,comp_num + 1);
a_param = zeros(q,1);
b_param = zeros(q,1);
k_param = zeros(q,1);
m_param = zeros(q,1);
sigma = zeros(q,1);
sigma_eq = zeros(q,1);
m_centered = zeros(q,1);

for i = 1:q
    % regression
[B,TSTAT,S2,VCV,VCV_WHITE,R2,RBAR,YHAT] = ols(dataM(:,i),eigenportfolio,1); %include↙
the constant
beta1(i,:) = transpose(B);
r2 = zeros(nrows,1);

r1 = dataM(:,i) - YHAT; %obtain the residual process
 % for open trades, add to the residual process
if status(i)== 0
    for j = 2:nrows
         r2(j) = sum(r1(1:j));
    end
    [n_r,m_r] = size(r3(:,i));
    r3(1:n_r,i) = zeros(n_r,1);
    r3(1:nrows,i) = r2;
end

if status(i)~= 0
r2 = r3(trade_open(i,day-1)+1:nrows + trade_open(i,day-1),i);
end

% run the ar model
[ar_coeff,LL,ERRORS] = armaxfilter(r2,1,1);


a_param(i) = ar_coeff(1,1);
b_param(i) = ar_coeff(2,1);
if b_param(i)> 0.99
    b_param(i) = 0.99;
end
k_param(i)= -log(b_param(i));

sigma(i) = sqrt(var(ERRORS)*2*k_param(i)/(1-b_param(i)^2));

sigma_eq(i) = sqrt(var(ERRORS)/(1-b_param(i)^2));
if sigma_eq(i)==0
    sigma_eq(i) = 0.05;
end
alpha(i) = beta1(i,1);
end

for i = 1:q
   m_param(i) = a_param(i)/(1-b_param(i))-mean(a_param)/(1-mean(b_param));
   s(i) = -m_param(i)/sigma_eq(i)-(alpha(i)/(k_param(i)*sigma_eq(i)));
```

```
end

beta = beta1(1:q,2:comp_num+1); %save all the betas from the regression
```

```
Rsigma = std(data_pca);
data1 = zscore(data_pca);
data2 = transpose(data1)*data1;
[U,S,V] = svd(data1);
sv = svd(data_pca);
pc_var = cumsum(sv.*sv)./sum(sv.*sv);
w = 0;
ncomp = 0;
[nnrow,nncol] = size(data_pca);

while ncomp == 0
    w = w +1;
    if pc_var(w) > 0.5   %desired explanatory power
        ncomp = w;
    end

end

Vm = V(:,1:ncomp);
```

```
for count = 241:f-242
data_market = data_in(count-239:count,:);
% new_matrix = zeros(40,40);
% n = 1;
% corr_matrix = corr(data_market);
% var_vector = var(data_market);
% sigmas = nthroot(var_vector,2);
ExpReturn = mean(data_market);
% for i = 1 :40
%     for j = 1:40
%         corr_matrix(i,j) = nthroot(corr_matrix(i,j), n);
%     end
% end
%ExpCovariance = corr2cov(sigmas, corr_matrix);
ExpCovariance = cov(data_market);
%StartPrice    = 100;
NumObs        = 1;
NumSim        = 1;
RetIntervals  = 1;     % one trading day
%NumAssets     = 5;
data_sim(count-240,:) = portsim(ExpReturn, ExpCovariance, NumObs, RetIntervals,↙
NumSim,'Expected' );
end
```

```
% clear all
% clc
clear all
clc
tic

s = warning('off', 'all');
filename = 'top40.txt';
data_in = importdata(filename);
[f,g] = size(data_in);
filename = 'jibar.txt';
ir2609 = importdata(filename);
int_rate = ir2609;
filename = 'index.txt';
index40 = importdata(filename);



count_run = 1;
fprintf('\nWrite data to a file, no screen output:\n');
fout = fopen('one_bootstrap1.txt','wt');
fprintf↙
(fout,'count_run\tx\tt_sim\tannualised_return\tannualised_cost\ttrade_fail\tprofit_vo↙
l\ttrade_count\n');

for x = 5:5:60
for t_sim = 1:20

row_max = f-x-1;

b_data= zeros(f,g);

for i = 1:x:row_max
    b_start = round(row_max*rand)+1; % random_draw
    b_end = b_start + x;
    b_data(i:x+i,:) = data_in(b_start:b_end,:);
end

b_data = b_data(1:row_max,:);




data = b_data;



   s_bo = 1.3;
   s_so = 1.2;
   s_bc = 0.85;
   s_sc = 0.95;
   pca_length = 240;
```

```
   exp_var = 0.5;
   reg_rows = pca_length;
   nrows = reg_rows;
   pca_rows = pca_length;
   start = max(nrows,pca_length);
   final = row_max - 1;

   main2

   fprintf(fout,'%d\t%.2f\t%.2f\t%.2f\t%.2f\t%.2f\t%.2f\t%.2f\n',count_run,x,t_sim,↙
annualised_return,annualised_cost,trade_fail,profit_vol,trade_count);
   count_run = count_run + 1;
end
end
fclose(fout);
toc
```